REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-98-

0808

Public reporting burden for this collection of information is estimated to average 1 hour per response, including th ... iering
and maintaining the data needed, and completing and reviewing the collection of information. Send comment ... on of
information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate fo ... Suite
1204, Arlington, VA 22202-4302, and to the Office of management and Budget, Paperwork Reduction Project (07C

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE December, 1995	3. REPORT TYPE AND D... Final

4. TITLE AND SUBTITLE
USAF Summer Research Program - 1995 High School Apprenticeship Program Final Reports, Volume 16, AEDC

5. FUNDING NUMBERS

6. AUTHORS
Gary Moore

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Research and Development Labs, Culver City, CA

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NI
4040 Fairfax Dr, Suite 500
Arlington, VA 22203-1613

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
Contract Number: F49620-93-C-0063

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for Public Release

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The United States Air Force High School Apprenticeship Program's (USAF- HSAP) purpose is to place outstanding high school students whose interests are in the areas of mathematics, engineering, and science to work in a laboratory environment. The students selected to participate in the program work in an Air Force Laboratory for a duration of 8 weeks during their summer vacation.

14. SUBJECT TERMS
AIR FORCE HIGH SCHOOL APPRENTICESHIP PROGRAM, APPRENTICEDHIP, AIR FORCE RESEARCH, AIR FORCE, ENGINEERING, LABORATORIES, REPORTS, SCHOOL, STUDENT, SUMMER, UNIVERSITIES

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

UNITED STATES AIR FORCE

SUMMER RESEARCH PROGRAM -- 1995

HIGH SCHOOL APPRENTICESHIP PROGRAM FINAL REPORTS

VOLUME 16

ARNOLD ENGINEERING DEVELOPMENT CENTER

RESEARCH & DEVELOPMENT LABORATORIES
5800 Uplander Way
Culver City, CA 90230-6608

Program Director, RDL
Gary Moore

Program Manager, AFOSR
Major David Hart

Program Manager, RDL
Scott Licoscos

Program Administrator, RDL
Gwendolyn Smith

Program Administrator, RDL
Johnetta Thompson

Submitted to:

AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
Bolling Air Force Base
Washington, D.C.
December 1995

PREFACE

Reports in this volume are numbered consecutively beginning with number 1. Each report is paginated with the report number followed by consecutive page numbers, e.g., 1-1, 1-2, 1-3; 2-1, 2-2, 2-3.

This document is one of a set of 16 volumes describing the 1995 AFOSR Summer Research Program. The following volumes comprise the set:

VOLUME	**TITLE**
1	Program Management Report
	Summer Faculty Research Program (SFRP) Reports
2A & 2B	Armstrong Laboratory
3A & 3B	Phillips Laboratory
4	Rome Laboratory
5A, 5B, & 5C	Wright Laboratory
6A & 6B	Arnold Engineering Development Center, Wilford Hall Medical Center and Air Logistics Centers
	Graduate Student Research Program (GSRP) Reports
7A & 7B	Armstrong Laboratory
8	Phillips Laboratory
9	Rome Laboratory
10A & 10B	Wright Laboratory
11	Arnold Engineering Development Center, Wilford Hall Medical Center and Air Logistics Centers
	High School Apprenticeship Program (HSAP) Reports
12A & 12B	Armstrong Laboratory
13	Phillips Laboratory
14	Rome Laboratory
15A&15B	Wright Laboratory
16	Arnold Engineering Development Center

HSAP FINAL REPORT TABLE OF CONTENTS i-xiv

STATIC FORCE AND MOMENT CALCULATION IN HYPERSONIC / SUPERSONIC GROUND TESTING

Phillip A. Chockley III

Shelbyville Central High School
401 Eagle Blvd.
Shelbyville, TN 37160

Final Report for:
High School Apprenticeship Program
Arnold Engineering Development Center

Sponsored By:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

and

Arnold Engineering Development Center

August 1995

STATIC FORCE AND MOMENT CALCULATION IN HYPERSONIC / SUPERSONIC GROUND TESTING

Phillip A. Chockley III
Shelbyville Central High School

Abstract

Both hypersonic and supersonic ground tests were studied. Testing involves placing scaled down aircraft such as space shuttles or missiles in a wind tunnel. Inside the tunnel, air is blown over the model at speeds ranging from Mach 1.5 to Mach 10. While wind is blowing, the model can be moved to many different positions which create different aerodynamic forces and moments. These forces and moments are measured by a balance located inside the model. The balance measures the forces and moments by monitoring changes in resistivity in the balance circuit and recording the change in electrical current which is converted in a term called raw counts (1638.4 counts per volt). Raw counts are converted into workable numbers in which, among other things, static force and moment calculations come from. These calculations are put into a Sixth Degree of Freedom (6DOF) database by the user to determine the stability and control characteristics of the test article.

STATIC FORCE AND MOMENT CALCULATION IN HYPERSONIC / SUPERSONIC GROUND TESTING

Phillip A Chockley III

Due to recent technological advances, the demand for new, more innovative aircraft has skyrocketed. Before companies can build these, their designs must first be tested in the form of a scaled down model in wind tunnels. One of the main purposes for running wind tunnel tests is to help engineers determine the stability and control of aircraft and missiles. The stability and control of a vehicle is determined from the aerodynamic forces and moments imposed on a vehicle at various flight attitudes. These forces and moments are measured by mounting a balance inside of a model that represents the actual flight vehicle and recording the balance measurements for predetermined flight attitudes.

The balance in the model measures six components. These components consist of two pitching moments, two yawing moments, a rolling moment, and a drag force. These components not only measure the force and moments resulting from aerodynamic loads, they also measure the forces and moments resulting from the model weight and it center of gravity. For the end user to determine the stability and control of the vehicle tested, the forces and moments resulting from the model weight and its center of gravity must be removed. The six balance components must be reduced into usable terms that can be used to determine the stability and control of a vehicle.

Changes in balance resistive are converted into forces and moments by multiplying the change in each balance gage resistive by a calibration value. These results are called balance components (BCn). The following is a definition for each BCn value:

BC1 - Pitching moment at x1
BC2 - Pitching moment at x2
BC3 - Yawing moment at x1
BC4 - Yawing moment at x2
BC5 - Rolling moment about balance center line
BC6 - Drag force

Each BC term includes aerodynamic forces and moments, as well as forces and moments created by the model and its center of gravity in relation to x1. These balance components are then used to calculate what is known as double prime terms. Double prime components consist of forces and moments relative to x1 in the balance axis.

The normal force double prime term (FN") is calculated by subtracting the moment measured at x2 (BC2) from the moment measured at x1 (BC1) which act in the in the x-z axis system, then dividing by the distance between x1 and x2. The equation used to define this term is:

FN"= (BC2-BC1)/(x1-x2)

The pitching moment double prime term (MM") is located a x1 and is equal to the BC1 term.

By once again subtracting the moment measured at x2 (BC4) from the moment measured at x1 (BC3), then dividing by the distance between x1 and x2, the side force (FY") can be calculated. The only difference, however, between normal and side force is that normal force is measured in the x-z axis system, while side force is measured in the x-y axis system. The equation for side force is:

FY"=(BC4-BC3)/(x1-x2)

BC3 not only helps calculate FY", but also completely defines yawing moment(MN"). The final two BC terms, BC5 and BC6, are equal to rolling moment(ML") and drag force(FA"), respectively.

Once the double prime terms are found, the next step toward determining the aerodynamic loads is to calculate what is known as single prime terms. Single prime terms only include the forces and moments resulting from aerodynamic affects. Therefore, forces and moments resulting from model weight and it's center of gravity must be removed. To accomplish this, several equations are used, in which three new terms are introduced: d, e, and f. These are trigonometric functions used to remove the model weight and it's respective moment acting at x1. Their values are calculated through a matrices operation using several variable angles and explained as weight vectors along each axis. Once calculated, these values are used in conjunction with double prime terms, model weight (W), and the model center of gravity in relation to x1 to attain the single primes. In the case of normal force, the model weight is multiplied by the weight vector along the z axis (f), then added to the double prime term. The following equation is used:

FN'=FN"+Wf

The pitching moment equation, as well as other single prime moment equations, incorporates yet another set of variables: the x-bar , y-bar, and z-bar. It is the model's center-of-gravity location relative to x1 in each respective axis. The pitching moment double prime term is added to the product of the model weight, the x-bar, and the weight vector along the z axis, and the product of the model weight, the z-bar, and the weight vector along the x axis is subtracted to calculate the pitching moment single prime value. The equation is:

$$MM'=MM''+Wxf-Wzd$$

Determining side and drag force is very similar to determining normal force. In each case the double prime term is used, as is model weight. In side force, model weight is multiplied by the weight vector along the y axis then subtracted from the double prime term, and in drag force, the model weight is multiplied by the weight vector along the x axis then added to the double prime term. Their equations are:

$$FY'=FY''-We$$

$$FA''=FA''+Wd$$

As in the forces, single prime moment equations are similar as well. Like pitching moment, combinations of model weight, weight vector, and center-of-gravity location is added and subtracted from the double prime term to find yawing and rolling moments. These are the equations:

$$MM'=MM''-Wxe+Wyd$$

$$MN'=MN''+Wze-Wyf$$

At this point the data is strictly in the balance axis system. In order for the end user to evaluate the aerodynamic data, the forces and moments in the balance axis system must be rotated into the model axis system. This is accomplished by knowing the pitch, roll, and yaw misalignment angles between the balance and model. The unprimed terms are calculated through yet another series of equations using a matrix operation that rotates the model's forces and moments into the model axis system. The resulting values are:

FN = normal force in the model axis system
MM = pitching moment in the model axis system
FY = side force in the model axis system
MN = yawing moment in the model axis system
ML = rolling moment in the model axis system
FA = drag force in the model axis system

The last step in force and moment calculation is the nondimensionalization of the unprimed terms. This constitutes eliminating the units of a value, leaving only the coefficient. This is accomplished in forces by dividing the unprimed term by dynamic pressure and reference area. In moments, it is accomplished by dividing the unprimed term by dynamic pressure, reference area, and reference length. By nondimensionalizing the calculations it makes it possible for the user to compare the data to that collected by other wind tunnels, or upscale the data to the size of the full scale object.

If not for the precision used in calculating forces and moments during wind tunnel testing, aircraft today would not have the ability to perform as well as they do. Through a series of mathematical equations, raw data is transformed into information the user can work with. With the knowledge a user gains from the wind tunnel, he can make changes in design flaws or simply change something he is unsatisfied with. Whatever the results may be, there is always something good to come from a wind tunnel test.

ASSESSMENT OF THE OBSCURATION MODEL IN THE STANDARDIZED INFRARED RADIATION MODEL (SIRRM)

Jennifer L. Counts

Franklin County High School
Dinah Shore Boulevard
Winchester, TN 37398

Final Report for:
High School Apprenticeship Program
Arnold Engineering Development Center

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, Washington, DC

and

Arnold Engineering Development Center

August 1995

ASSESSMENT OF THE OBSCURATION MODEL IN THE STANDARDIZED INFRARED RADIATION MODEL (SIRRM)

Jennifer Counts
Franklin County High School

Abstract

The efficient design of airborne infrared missile warning receivers depends upon the ability to compute the infrared radiation incident upon the sensor in typical engagement scenario. Missile warning receivers are designed to detect the infrared radiation given off by the plume of the missile. The most typical engagement scenario is when the missile is approaching the aircraft from the rear. This rear approach causes the sensor to detect the plume radiation from nose on. From the nose on view, a portion of the missile plume is obscured by the missile body. It is important to know how much radiation is received by the sensor in order to ensure the most efficient design of the infrared missile warning receiver. The Standardized InfraRed Radiation Model (SIRRM) is the Joint Army Navy NASA Air Force (JANNAF) standard model for computing the received radiation.

In this project I tested the accuracy of the obscuration model in SIRRM. Since SIRRM deals only with axi-symmetric objects and the analytical solutions for them were not straight forward, the SIRRM results could not be directly compared to the analytical results. A Raytracing Code, which deals with both axi-symmetric and non-axi-symmetric objects, was adapted from a previously existing code. This was done in order to have valid results to compare to the SIRRM results. Two analytical cases and their solutions were found in order to test the accuracy of the Raytracing Code. Finding the Raytracing Code to be accurate, six objects were developed to run in SIRRM. After plotting and comparing the SIRRM results and the Raytracing Code results, it was discovered that SIRRM has a maximum five percent error in the obscuration at small aspect angles.

ASSESSMENT OF THE OBSCURATION MODEL IN THE STANDARDIZED INFRARED RADIATION MODEL (SIRRM)

Jennifer L. Counts

Introduction

> Development of early warning strategic missile surveillance systems and tactical missile warning receivers depends in part on the reliable prediction of infrared radiation from rocket exhaust plumes using target signature models. These models must be capable of treating a wide range of propulsion and trajectory parameters for threat missile systems, as well as sensor variables such as spectral band pass, and ***viewing geometry***. (emphasis added)

(Excerpted from AFRPL-TR-81-54 STANDARDIZED INFRARED RADIATION MODEL (SIRRM) Volume 1: Development and Validation, Reference 1)

The efficient design of airborne infrared missile warning receivers depends upon the ability to compute the infrared radiation incident upon the sensor in typical engagement scenarios. Missile warning receivers are designed to detect the infrared radiation given off by the plume of the missile. The most typical engagement scenario is when the missile is approaching the aircraft from the rear. This rear approach causes the sensor to detect the plume radiation from nose on. From the nose on view, a portion of the missile plume is obscured by the missile body. It is important to know how much radiation is received by the sensor in order to ensure the most efficient design of the infrared missile warning receiver. Standardized InfraRed Radiation Model (SIRRM) is the Joint Army Navy NASA Air Force (JANNAF) standard model for computing the received radiation. In this project I tested the accuracy of the obscuration model in SIRRM.

Acknowledgments

I would like to thank Robert S. Hiers III for his time, patience, encouragement, and help throughout the project, Martha Simmons for her assistance with the XDAT computer program, David Pruitt for his help and encouragement on this project, Rick Roepke and Bonnie Heikkinen for their help with various computer applications, Danny Brown for his help with the Raytracing Codes, and the personnel of ES5 Plume Data Center, EL3 Propulsion Diagnostics, and EL2 Propulsion Computational Technology Section for their encouragement and assistance in this project.

Methodology

The total brightness of a missile plume is proportional to the visible plume volume for optically thin and uniform plumes. Therefore, the computation of total brightness is really only the computation of visible plume volume. This turns the project into a geometry problem. The situation consists of one body (the missile body) obscuring another body (the plume). The solution is to find how much of the plume volume is visible as a function of the viewing angle (See Figure 1). SIRRM will compute this for a cylindrical missile body with a general plume. I would have preferred to compare the SIRRM results directly to analytical solutions; however, this was not feasible, since SIRRM only deals with axi-symmetric objects. An example of an axi-symmetric object is a cylindrical missile body and a conic plume. Since the analytical solutions for axi-symmetric objects were not straight forward, a raytracing program was written which would compute these solutions. In addition, it would also compute the solutions for general non-axi-symmetric objects. This program was adapted from a Raytracing program already in existence (Reference 2). Since a particular class of non-axi-symmetric objects possesses analytical solutions, these solutions were found and compared to the Raytracing results. This validates the Raytracing Code as a viable bench mark to which other non-analytical solutions can be compared. The Raytracing results were then to be compared to the SIRRM results, thus assessing the accuracy of SIRRM.

In order to validate the Raytracing Code as a bench mark for non-axi-symmetric objects, solutions for an analytical case were calculated. Using a diagram of a right rectangular prism obscured by a right rectangular prism (See Figure 2), the following equations for volume were derived:

$$V_{total} = d_2 L$$

$$V_{obscured} = 0.5 \frac{d_3}{\tan\vartheta}$$

$$V_{visible} = V_{total} - V_{obscured}$$

$$\frac{V_{visible}}{V_{total}} 100 = \text{Percent Visible}$$

As noted in the diagram, the optimum solution is not always found by the indirect method of calculating the visible volume by subtracting the obscured volume from the total volume. In some cases, it is better to directly calculate the visible volume. I assigned theta critical (ϑ_c) to the angle at which the transition from the indirect calculation to the direct calculation of the visible volume takes place. The two equations for the different cases of $\frac{V_{visible}}{V_{total}}$ are stated in the following:

$For\ 0 \le \vartheta \le \vartheta_c$

$$\frac{V_{visible}}{V_{total}} = 0.5\frac{L}{d}\tan\vartheta$$

$For\ \vartheta_c \le \vartheta \le \frac{\pi}{2}:$

$$\frac{V_{visible}}{V_{total}} = 1 - 0.5\frac{d}{L\tan\vartheta}$$

I used these equations to find the analytical solutions for three prism obscured by a prism cases. Each of the three cases was constructed a different size. These cases included a length divided by depth (L/d) of one, five, and ten. A table was made for each case using angles from zero through ninety degrees, theta critical, and their calculated solutions. Using XDAT (Reference 3), a computer program that creates graphs, I plotted the solutions of the three cases for comparison with the Raytracing Code results. I derived the X, Y, and Z coordinates for a right rectangular prism and used the points to construct the prism out of triangles. This technique was used in order to draw the figure in FAST (Reference 4). FAST is a computer program which takes the information supplied and creates a drawing called a wireframe. FAST requires the following information in this order:

A. Number of points, Number of triangles, Number of tetrahedrons
B. X-coordinates, Y-coordinates, Z-coordinates
C. Vertices of the triangle
D. Surface number of the object

A wireframe was generated from the information given to FAST. Using the FORTRAN computer language, a code was written in order to supply FAST the information needed to create the objects.

Another analytical case was created and solved. This case was constructed of a pyramid obscured by a prism (See Figure 3). As before, theta critical (ϑ_c) was assigned to the angle at which the transition from the indirect calculation to the direct calculation of the visible volume took place. This case, however, was also assigned a theta double critical (ϑ_{cc}). The different equations are as follows:

$For\ \vartheta \rangle \vartheta_c$ and $\vartheta \rangle \vartheta_{cc}$

$$\frac{V_{visible}}{V_{total}} = 1 - \frac{0.5dA\cos\phi}{L_2^2\tan\phi + dL_2}$$

For $\vartheta \ \langle \ \vartheta_c$ and $\vartheta \ \langle \ \vartheta_{cc}$:

$$\frac{V_{visible}}{V_{total}} = \frac{0.5L_{11}L_{12}\sin(\phi - \vartheta) + 0.5L_3L_{14}\cos\vartheta}{L_2^2 \tan\phi + dL_2}$$

For $\vartheta \ \rangle \ \vartheta_c$ and $\vartheta \ \langle \ \vartheta_c$

$$\frac{V_{visible}}{V_{total}} = \frac{0.5L_3L_{14}\cos\vartheta}{L_2^2 \tan\phi + dL_2}$$

For $\vartheta = 0$:

$$\frac{V_{visible}}{V_{total}} = 1 - \frac{dL_2}{L_2^2 \tan\phi + dL_2}$$

Analytical solutions for some representative cases are shown in Figure 4.

Six cases were created for the assessment of the Raytracing Code. The first three were a prism obscured by a prism. The length was the same for every object, but the height and the width values changed from one to one-fifth to one-tenth. The other three cases were a pyramid obscured by a prism. The prism was constructed with a length, height, and depth of five, and the pyramid had a length of fifty, a depth of five, and a half-angle of fifteen, thirty, and forty-five degrees. These cases were combined using a code written in FORTRAN.

The Raytracing Code was the source I used to test the accuracy of SIRRM. The Code's primary function is to shoot rays through an object from a given point (See Figure 5). The program consists of a "block" that is made of many points from which the line of sight rays are shot. The program determines which facet, or triangle, the line of sight ray passes through and gives the X, Y, and Z coordinates of the point of intersection. I took the information provided by the Raytracing Code results and wrote a code to interpret them and determine the ratio of visible volume to total volume. This code first determined whether the point of intersection was on the plume or the obscuring body. It took the points and calculated the lengths from one point to the next on a certain line of sight. Running sums of the visible lengths of the lines of sight through the plume and the total lengths of the lines of sight through the plume were tabulated as shown below:

$$\sum_{visible} L_i \qquad\qquad \sum_{total} L_i$$

It is possible to assign every point in the Raytracing "block" a certain base area (A_b)and to calculate the total volume swept out by every line of sight. This calculation allows a numerical value to be assigned to both the total volume and the visible volume. Since I was only interested in the ratio of visible volume to total volume, this procedure was not necessary. If "J" is assigned to represent the brightness of the plume, and $J_{(90°)}$ is the brightest the plume can be (since, at 90° aspect angle, the plume is completely unobscured by the missile body), the following equations show the relationship between the ratio of the visible volume to the total volume and the ratio of the brightness of the plume at a certain angle to the brightness of the plume at ninety degrees:

$$\frac{V_{visible}}{V_{total}} = \frac{A_b \sum_{visible} L_i}{A_b \sum_{total} L_i} = \frac{\sum_{visible} L_i}{\sum_{total} L_i} = \frac{J_{(\vartheta)}}{J_{(90^o)}}$$

Since A_b appears in both numerator and denominator, it can be canceled. This equation also shows that the visible volume is proportional to the summation of the visible lengths, and the total volume is proportional to the summation of the total lengths. Thus, the ratio of the visible volume to the total volume calculated from the Raytracing Code information could be compared to the brightness ratio calculated by SIRRM, thus providing an adequate test of the SIRRM accuracy.

The Raytracing Code was run for the six analytical cases for angles zero through ninety degrees, and the results recorded. I found that the analytical solutions and the Raytracing results matched very closely (See Figure 6) with errors between the Raytracing results and the analytical solutions within one-half percent. This error was found at the smaller aspect angles in all the cases. This conclusion was confirmed when a plot was generated in XDAT showing the Raytracing results versus the analytical solutions (See Figure 7).

Finding the Raytracing results to be true and the Raytracing Code to be accurate, six axi-symmetric cases were created in order to compare their Raytracing results to the SIRRM results, thus assessing the accuracy of SIRRM. The cases were constructed using cylinders and frustrums (See Figure 8). One cylinder was used to represent the obscuring missile body, and the frustrum and other cylinder to represent the missile plume. A code was created using FORTRAN to generate, under given conditions, the information FAST required and to put it in a file that FAST would read. This code was generalized in order to allow the construction of cylinders and frustrums of different sizes. For the objects I would be using in my tests, the obscuring cylinder was always constructed by the same dimensions. It was made with a length of five and a radius of one. The frustrum and the cylinder used to represent the plume, on the other hand, had three different dimensions. This was accomplished by changing the length of the radius and the value of the half-angle in the FORTRAN code. In all the cases, the length of the frustrums and the cylinders representing plumes was twenty. The radius changed from one-half to one to two, and for the frustrum, the half-angle changed from fifteen to thirty to forty-five degrees. The half-angle of the cylinder remained at zero. Another code, also using FORTRAN, was created to combine the cylinder with the three frustrums and the cylinder with the three cylinders, thus creating the test cases for SIRRM. For the six cases, I calculated the percent visible at zero degrees using the equations derived for the right rectangular prism cases. This provided a comparison for the Raytracing Codes, since the analytical solutions could not be directly calculated. A code was created to calculate the percent visible under given conditions for all angles zero through ninety degrees. The Raytracing Code was then run for all six cases and the results placed in files for later use.

I created files for the six axi-symmetric cases to receive the results of the SIRRM runs. The SIRRM test run resulted in major errors between the Raytracing Code results and the SIRRM results. I had to increase the number of points and the lengths of the objects in order to get better results. The fact that SIRRM was sensitive to the size of the object was my first conclusion. After changing the dimensions in all the SIRRM cases, the first SIRRM run was executed in all cases using aspect angles from zero to ninety, concentrating on near nose aspects. The results were placed in the previously created files.

I then tested the accuracy of SIRRM further by evaluating the six cases without the obscuring cylinder body. Since an aspect angle of ninety degrees shows no obscuration and the calculation was known, I could compare the calculation with the new results. The value at ninety degrees from the first SIRRM run and the value at all the angles in the second SIRRM run should have been equal. These results were placed in the same files with the first SIRRM results in order to see the difference in the two numbers.

The results for a cylindrical plume obscured by a cylindrical missile body computed by SIRRM and by the Raytracing code are compared in Figure 9. For this cases, the plume of a larger diameter than the missile body. Percent errors between the Raytracing Code results and the first SIRRM results ranged from five percent at the small aspect angles, to one-tenth percent at the larger aspect angles (See Figure 10). Calculated errors between the results of the two SIRRM runs indicated a maximum error slightly greater than four percent at the smaller aspect angles. These plots gave the final conclusion that SIRRM has an approximate five percent error in its calculations, especially in the smaller aspect angles.

Conclusions

The most important conclusion of the project was that SIRRM had a maximum error of five percent, occurring only in the smaller aspect angles. Since the number of points and the lengths of the test objects had to be increased, the second conclusion was that SIRRM is very sensitive to the size of the object. This means that it is possible for a small object being tested to have worse results than a geometrically similar but larger object being tested. The fact that the Raytracing Code is always accurate in its calculations was another conclusion. In the two analytical cases, there was less than one-half percent error between the Raytracing Code results and the analytical solutions at all the angles.

Recommendations

My project dealt only with optically thin cases, where the brightness is proportional to the volume. In optically thick cases, the brightness is proportional to the projected surface area. The Raytracing Code could be revised to calculate the projected surface area, therefore enabling the obscuration model in SIRRM to be further tested. This is a suggestion for further work in order to test the obscuration model in SIRRM for another limiting case where the computation of brightness is really a geometry problem.

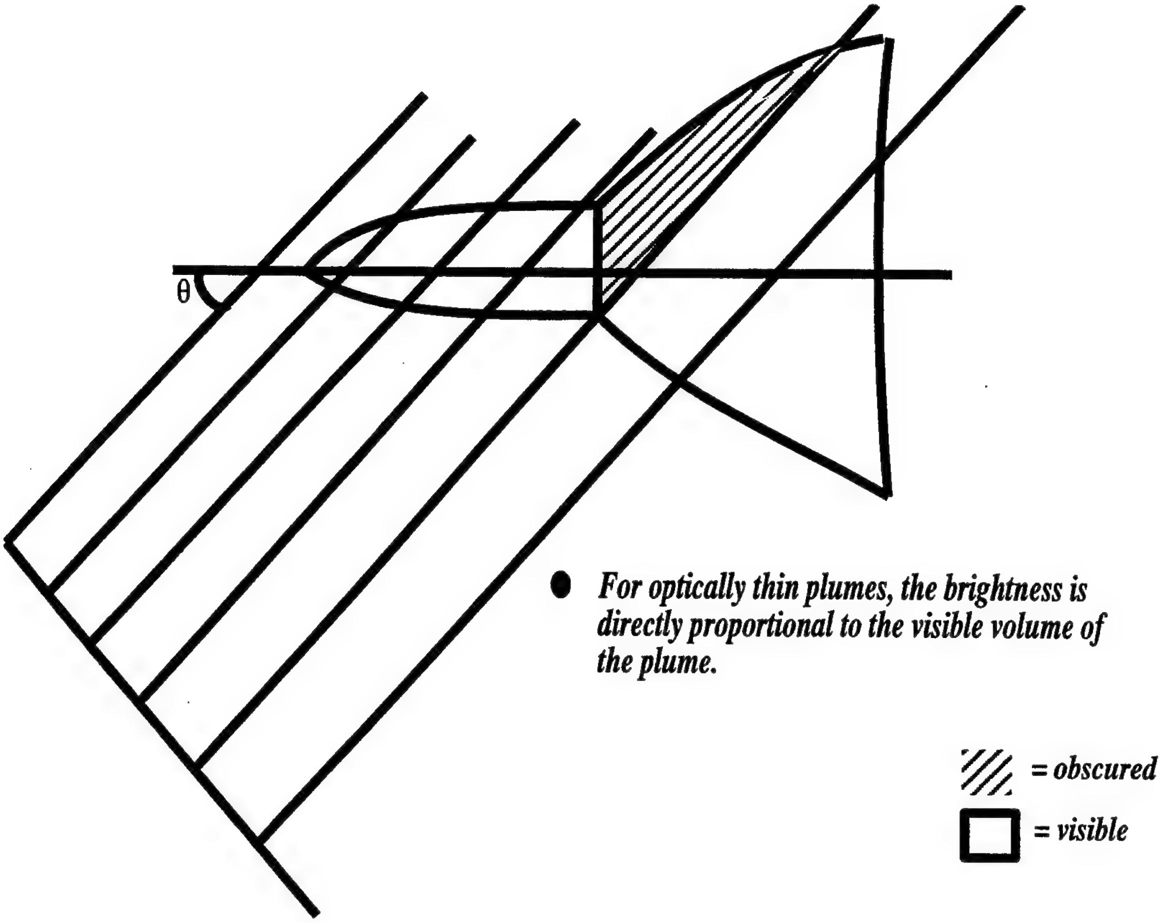
References

1. Ludwig, C. B., et al," Standardized Infrared Radiation Model (SIRRM), Volume 1: Development and Validation," AFRPL-TR-81-54.

2. Sverdrup Technology Memo, R. S. Hiers III to H. T. Bentley, III, 19 July 1993.

3. Holt, D., " XDAT Users Manual," Unpublished Sverdrup Technology Document.

4. Walatka, P. P., " FAST User Guide," COSMIC PROGRAM # ARC-13316, NASA Ames Research Center, October 1993.



- ***For optically thin plumes, the brightness is directly proportional to the visible volume of the plume.***

Figure 1 The General Problem

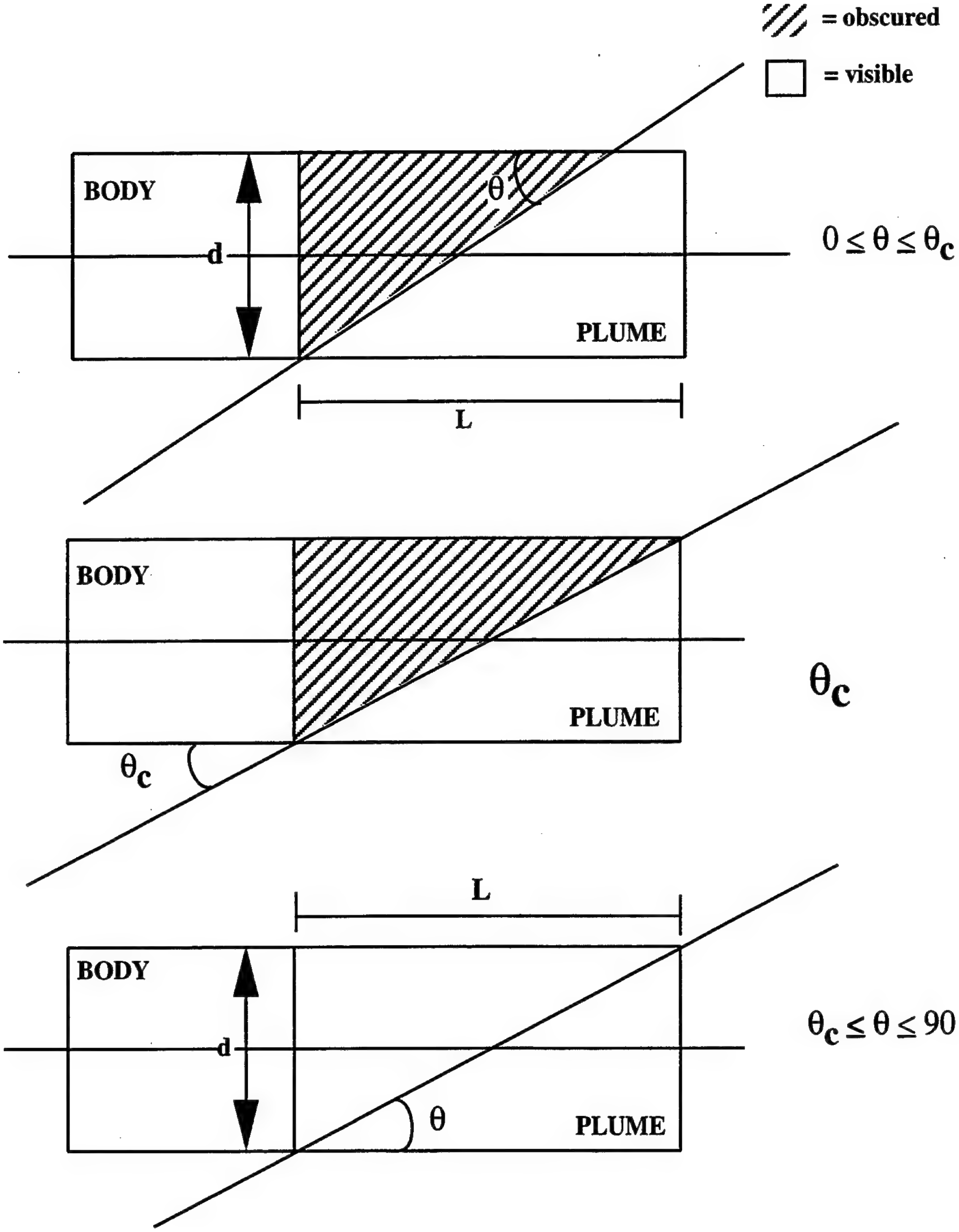


Figure 2 Analytical Solutions

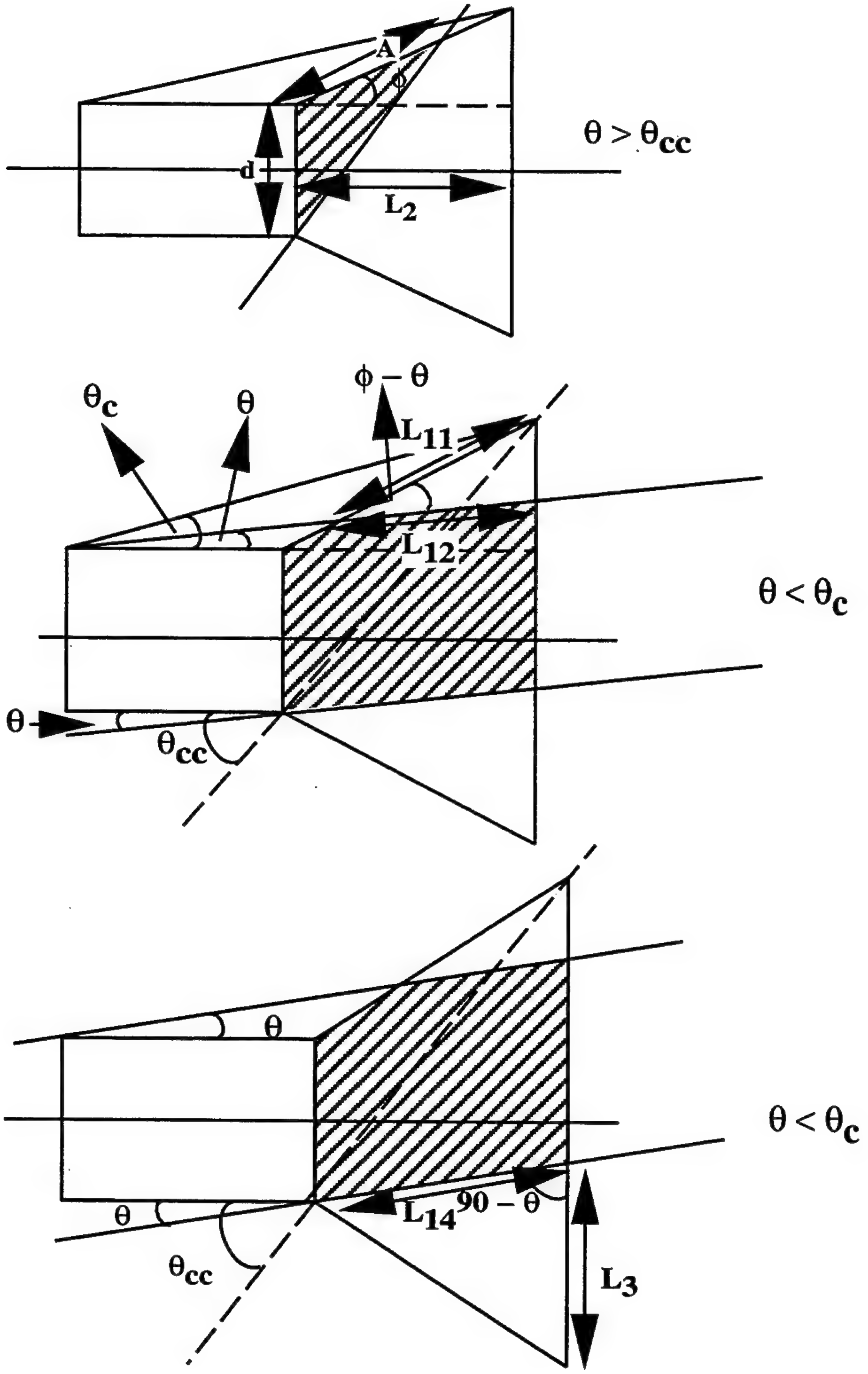


Figure 3 Analytical Solutions

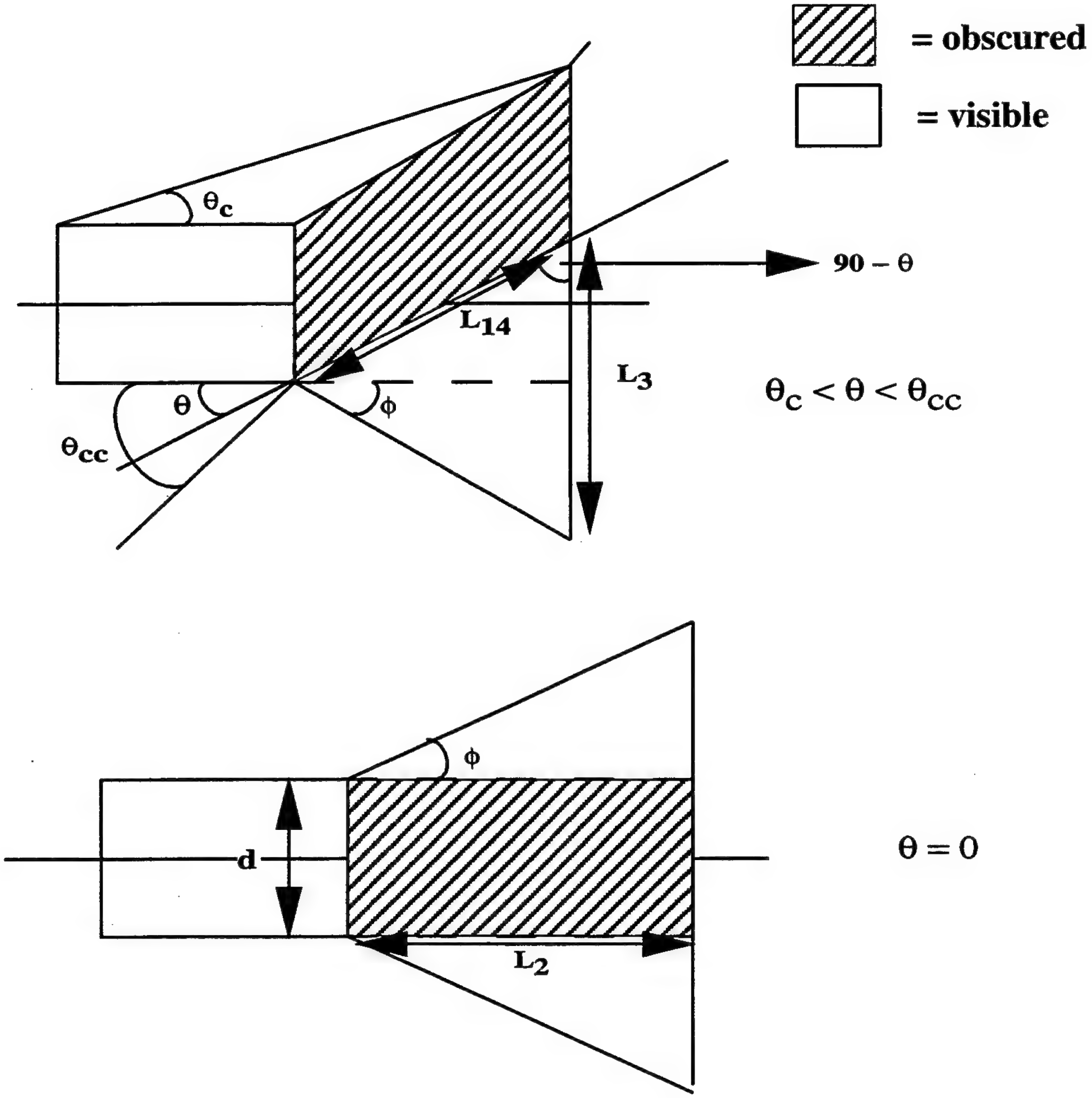


Figure 3 Analytical Solutions (Concluded)

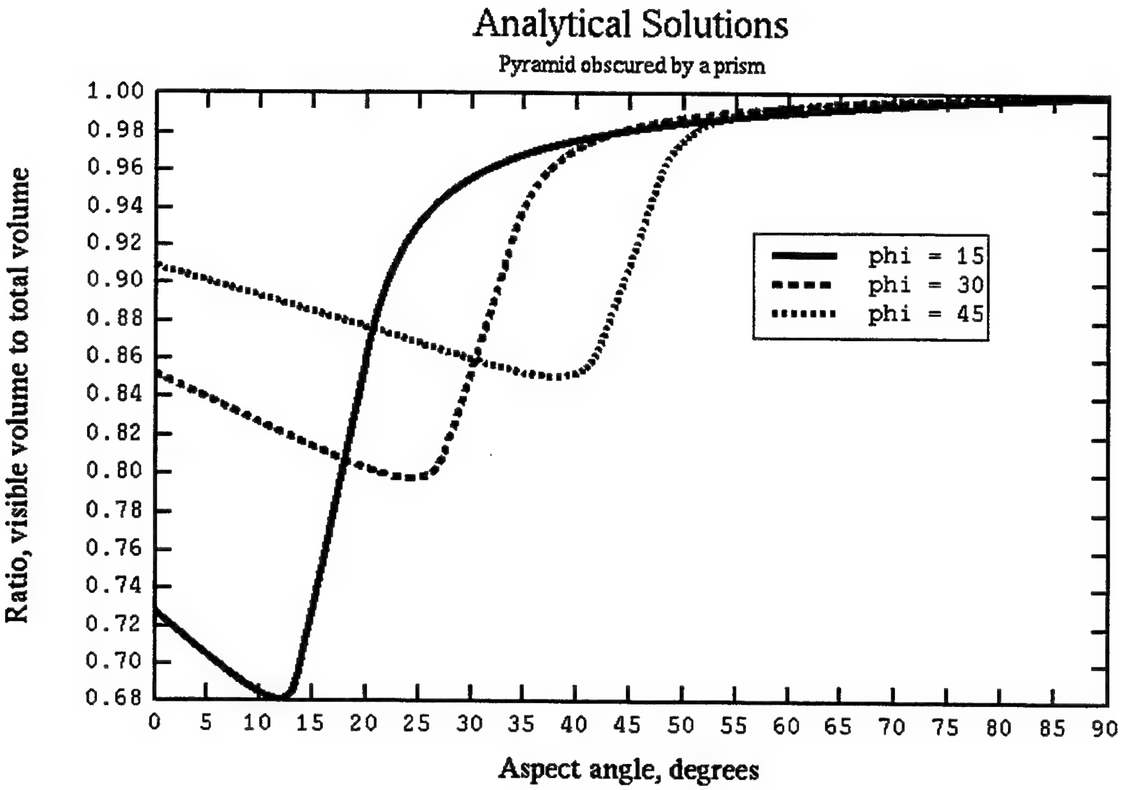


Figure 4 Analytical Solutions for Pyramids Obscured by Prisms

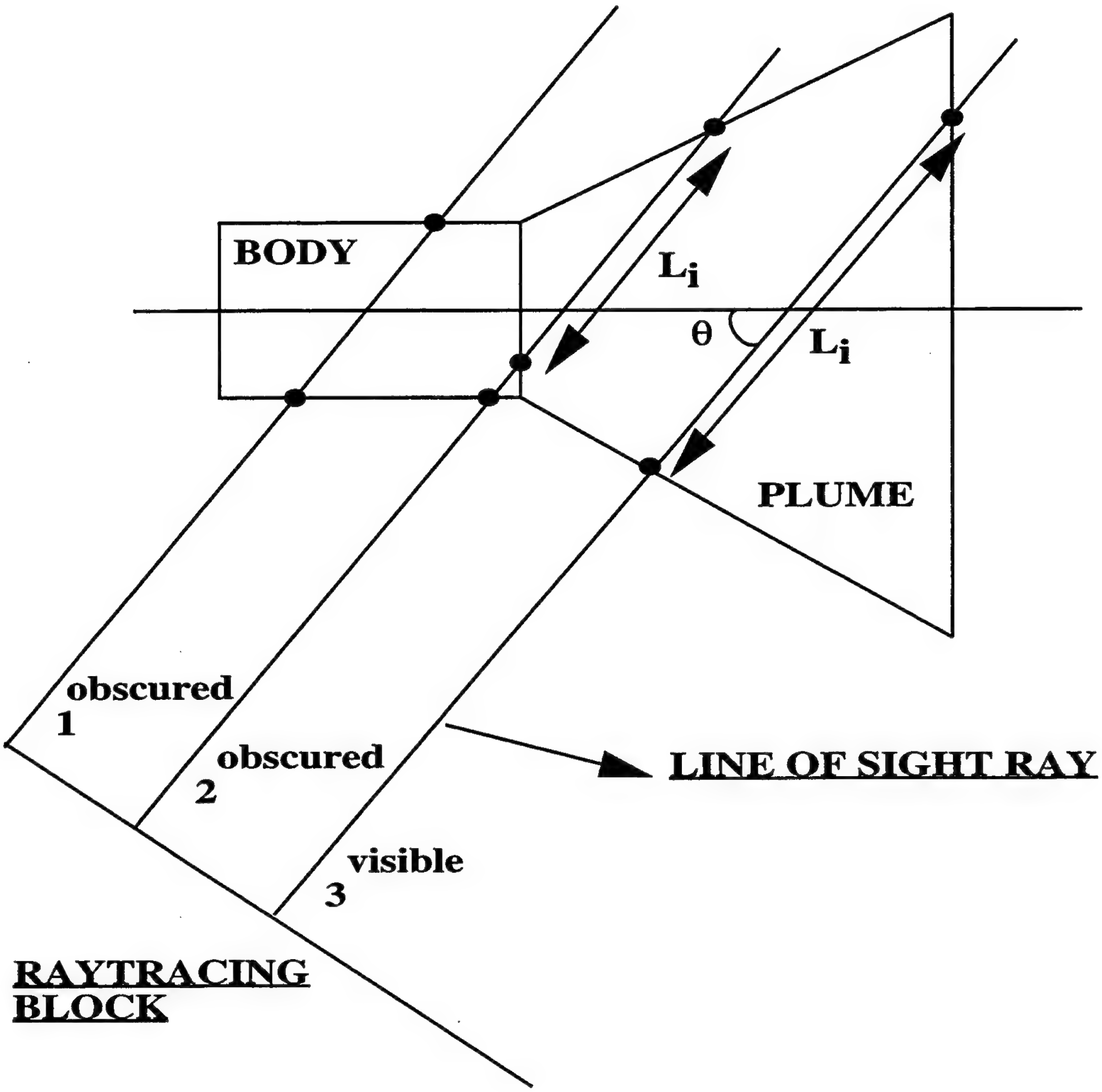


Figure 5 Raytracing Explanation

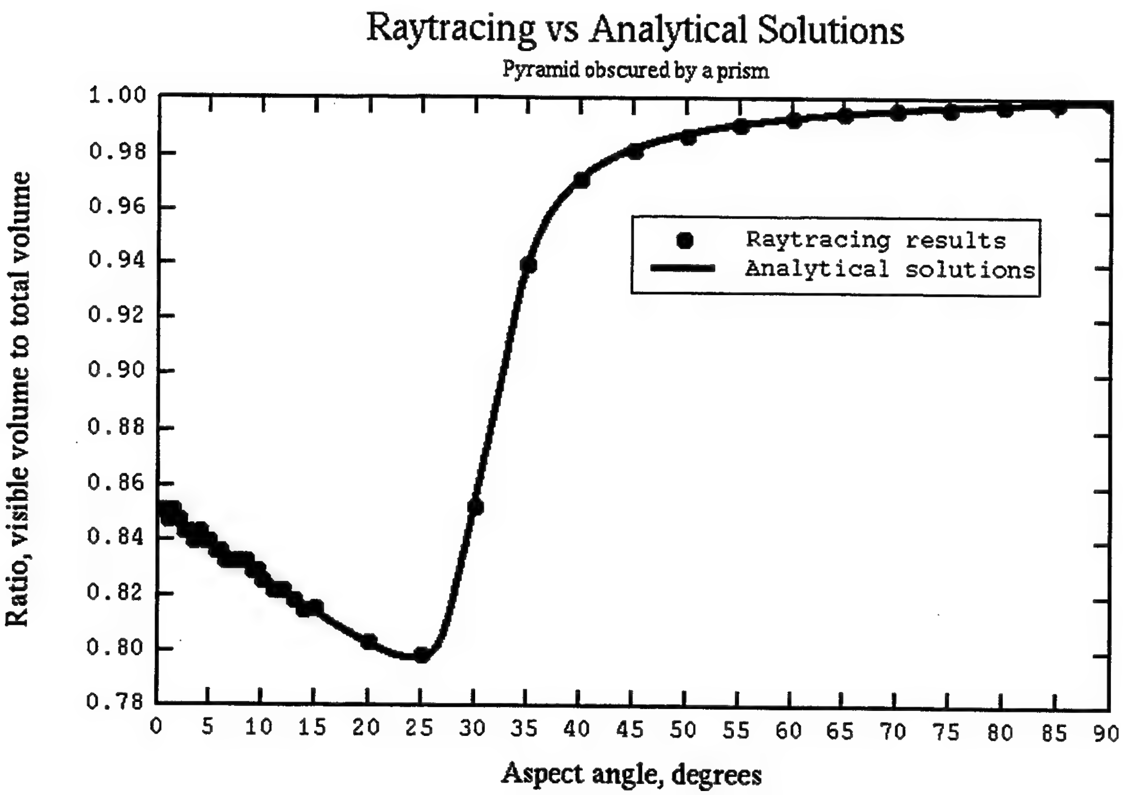


Figure 6 Raytracing vs Analytical Solutions for Pyramid Obscured by Prism

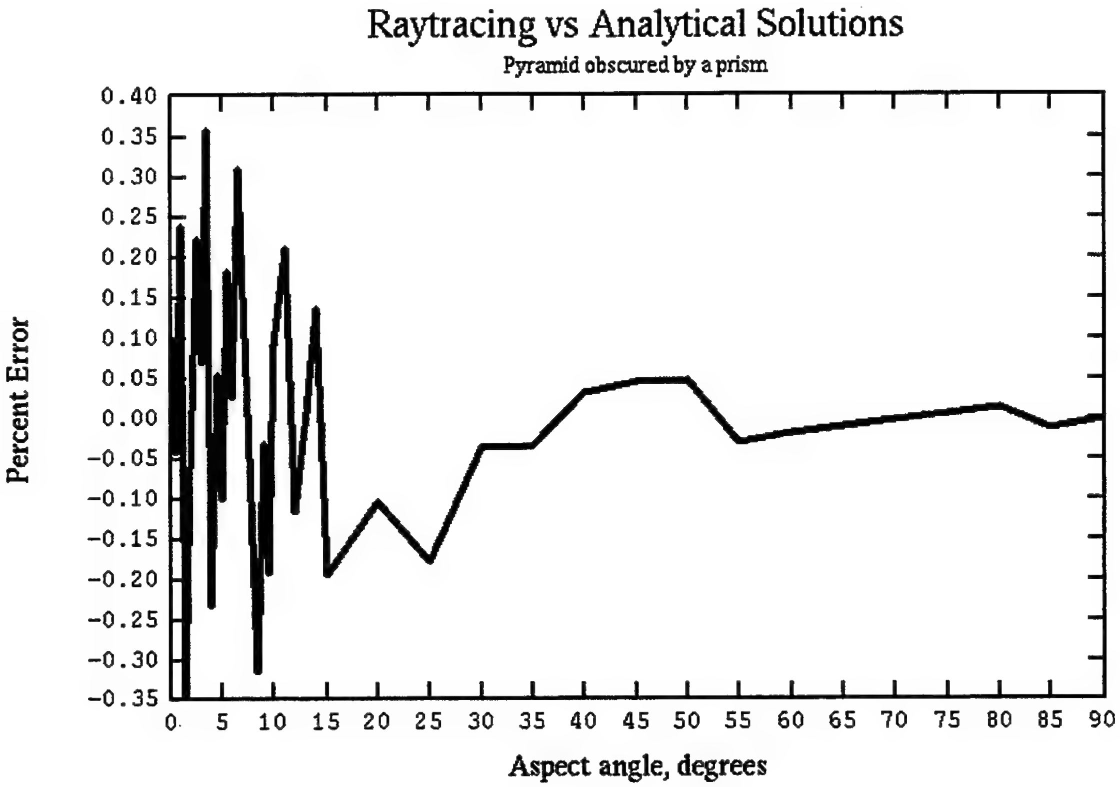


Figure 7 Error in Raytracing for Pyramid Obscured by Prism

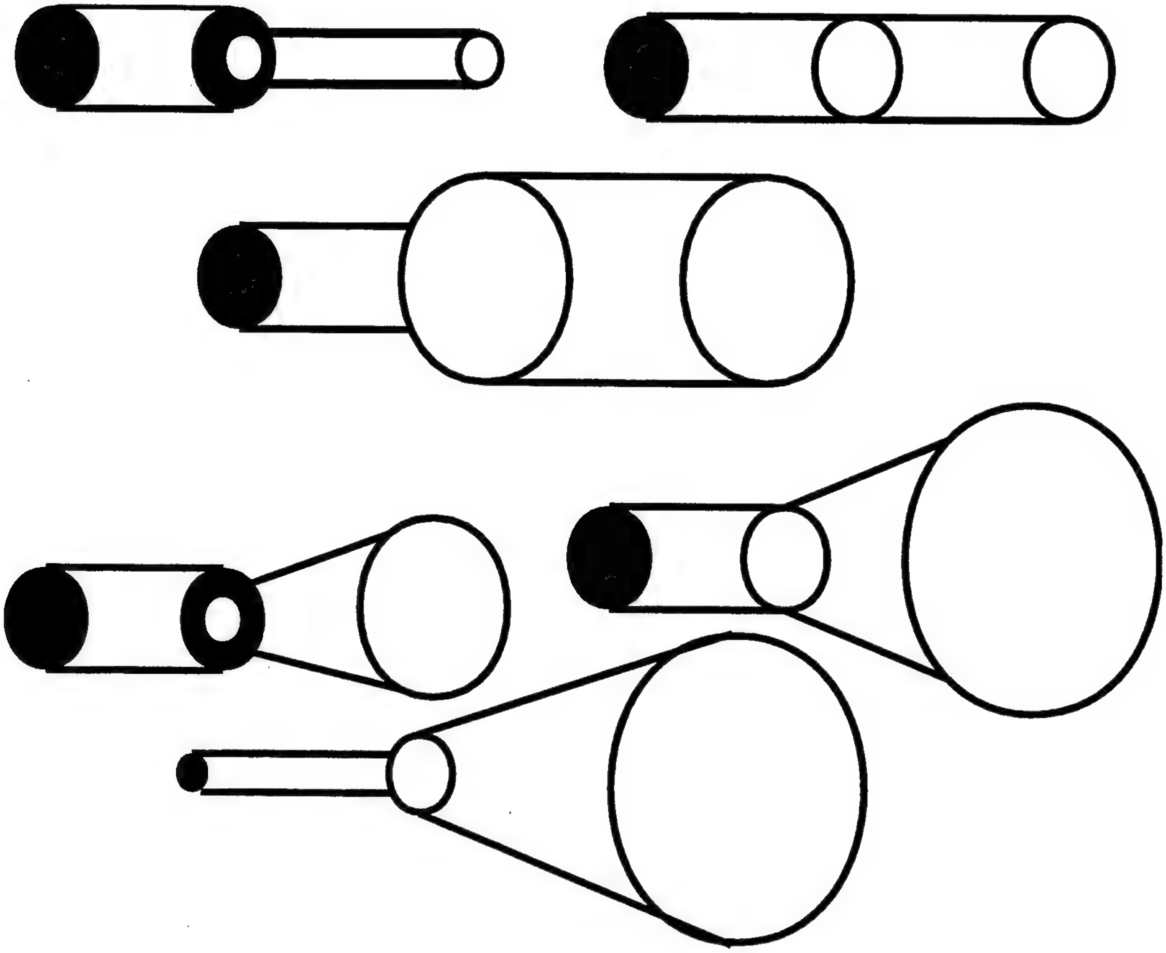


Figure 5 SIRRM Test Cases

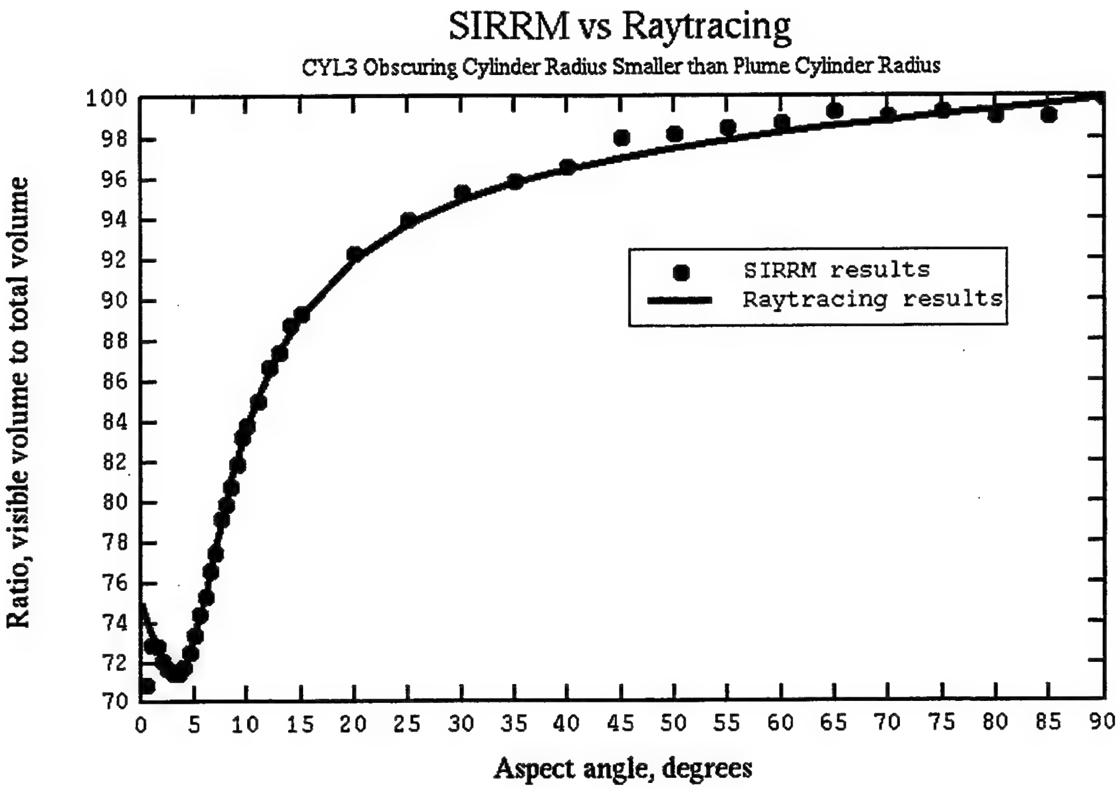


Figure 9 SIRRM vs Raytracing for Cylinder Obscured by Cylinder

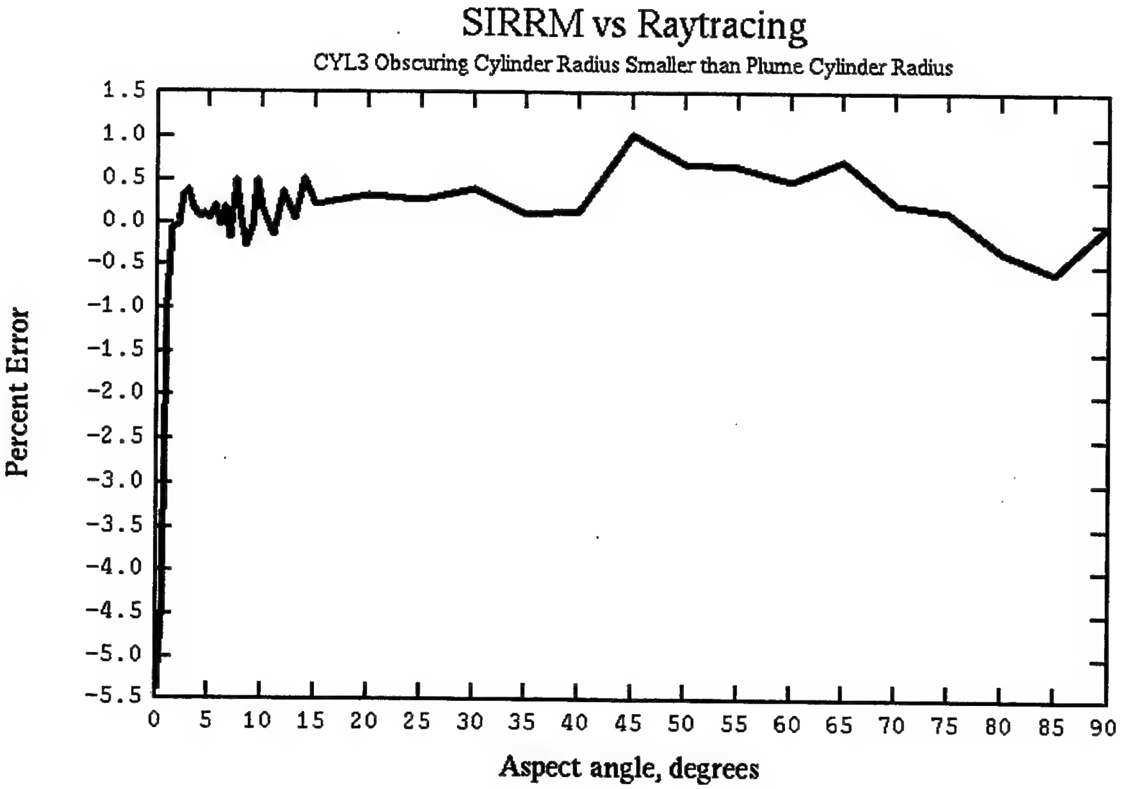


Figure 10 Error in SIRRM for Cylinder Obscured by Cylinder

Implementation of Network Diagramming Software

Michael L Fann

Tullahoma High School
1001 North Jackson Street
Tullahoma, TN 37388

Final Report for:
High School Apprentice Program
Arnold Engineering Development Center

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

and

Arnold Engineering Development Center

August 1995

Implementation of Network Diagramming Software

Michael Fann
Tullahoma High School

Abstract

ClickNet, a network diagramming software package, was implemented for use in analyzing the design and management of the OIS (Office Information Systems) unclassified ethernet network at Arnold Engineering Development Center. The software package was used to create a multiple level diagram of the network. The images in the diagram were linked to a database providing network information. This software package made it possible to replace the existing database with a graphical representation of the network connectivity and network device layout of each building. This format made analysis of connectivity and network supplies much easier. The design also made it simple to add network devices to the diagram and to easily change connectivity. After implementation of this package, the graphical representation of the network layout and supplies will be much more affective in managing and analyzing the OIS unclassified network. The design also allows connectivity and other network information to be shown at the same time, rather than in different databases. The ease in administration of the program with the multiple level network diagrams and floor plans and the ability to easily gain more information about network connectivity, network supplies, and network device layout prove the advantages of the implementation of this network diagramming software.

Acknowledgments

I would like to thank the Air Force Office of Scientific Research for giving me the opportunity to participate in this program. I would also like to thank James Mitchell for his involvement and leadership in the program at Arnold Engineering Development Center. Finally, I am especially thankful to Kemp Brooks, my mentor, and Rob Mathis for the assitance and guidance they gave me throughout my summer.

Implementation of Network Diagramming Software

Michael Fann

Introduction

ClickNet, a network diagramming software package, was purchased to be used in the analysis and management of networks at Arnold Engineering Development Center. Since the software had not been previously used at this facility, it had to be tested before being implemented for use in the OIS (Office Information Systems) unclassified ethernet network. Records for network supplies and device locations were previously documented on various databases. With implementation of this software, connectivity of the network can be shown graphically with layered drawings giving network information on devices such as: terminals, workstations, concentrators, and cables. Although, this program will give the same information as the existing databases, a graphical representation of connectivity and device layout will also be shown, making it easier and more functional for use.

Discussion of Problem

ClickNet must be implemented for use as analysis and management software for the OIS unclassified ethernet network at Arnold Engineering Development Center (AEDC). After testing it is to be determined if ClickNet Version 2.0 should be incorporated for network analysis and management at AEDC.

Methodology

Initially, a large inventory was taken of network supplies in many buildings connected to the OIS (Office Information Systems) unclassified ethernet network. The results from this inventory were entered into a Microsoft Excel spreadsheet. This inventory was taken to update and correct existing network supplies databases. Eventually the updated databases would be linked to the design created in ClickNet to give network information on each device. This experience also allowed me to become familar with the network supplies at Arnold Engineering Development Center and to learn more about the connectivity of each building on the network. After the inventory was completed, ClickNet Version 2.0, a newly purchased network diagramming software package, was examined for use for analysis and management of networks. It was determined ClickNet would satisfy the requirements needed for implementation in this capacity. ClickNet was then used to develop a multiple layered graphical database for management of the OIS unclassified network at Arnold Engineering Development Center.

Development of Multiple Layered Graphical Database

ClickNet was used to create a network inventory database which can be accessed from a graphics environment, demonstrating layouts of buildings and connectivity of the OIS unclassified network. A connectivity diagram (Figure 3-6) was designed as the initial layer of the database. This diagram displayed the connectivty of every building connected to the OIS unclassified network and distinguished each Hub building for the network. From this initial layer of the program, a user would be able to choose any building on the network to view network information.

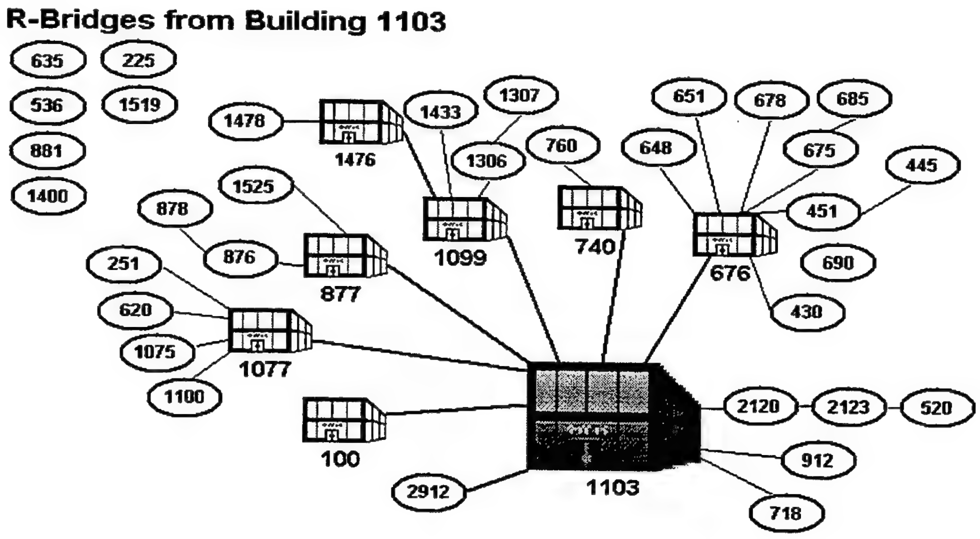


Figure 3-6

Secondary levels were designed by creating floor plans in Floorplan2.0, an architectural and space planning program. These floor plans were designed for each Hub building and made as secondary levels for the program. See Figure 3-7 for an example of a secondary level floor plan. Network images such as: terminals, concentrators, and cables were placed in the appropriate locations of each building.

Building 1103 (EAF) Second Level Floor Plan

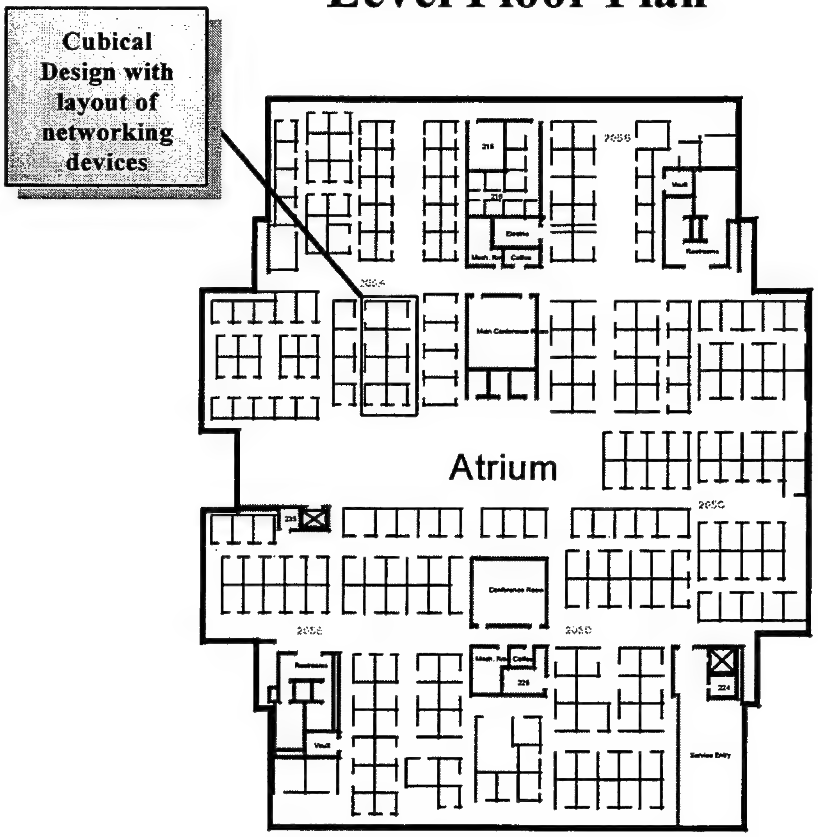


Figure 3-7

Network information was assigned to each image by connecting the images to an existing network inventory database. The network address field was completed for each image and used to connect to the existing database. An example of the network information given for each image can be seen in Figure 3-8.

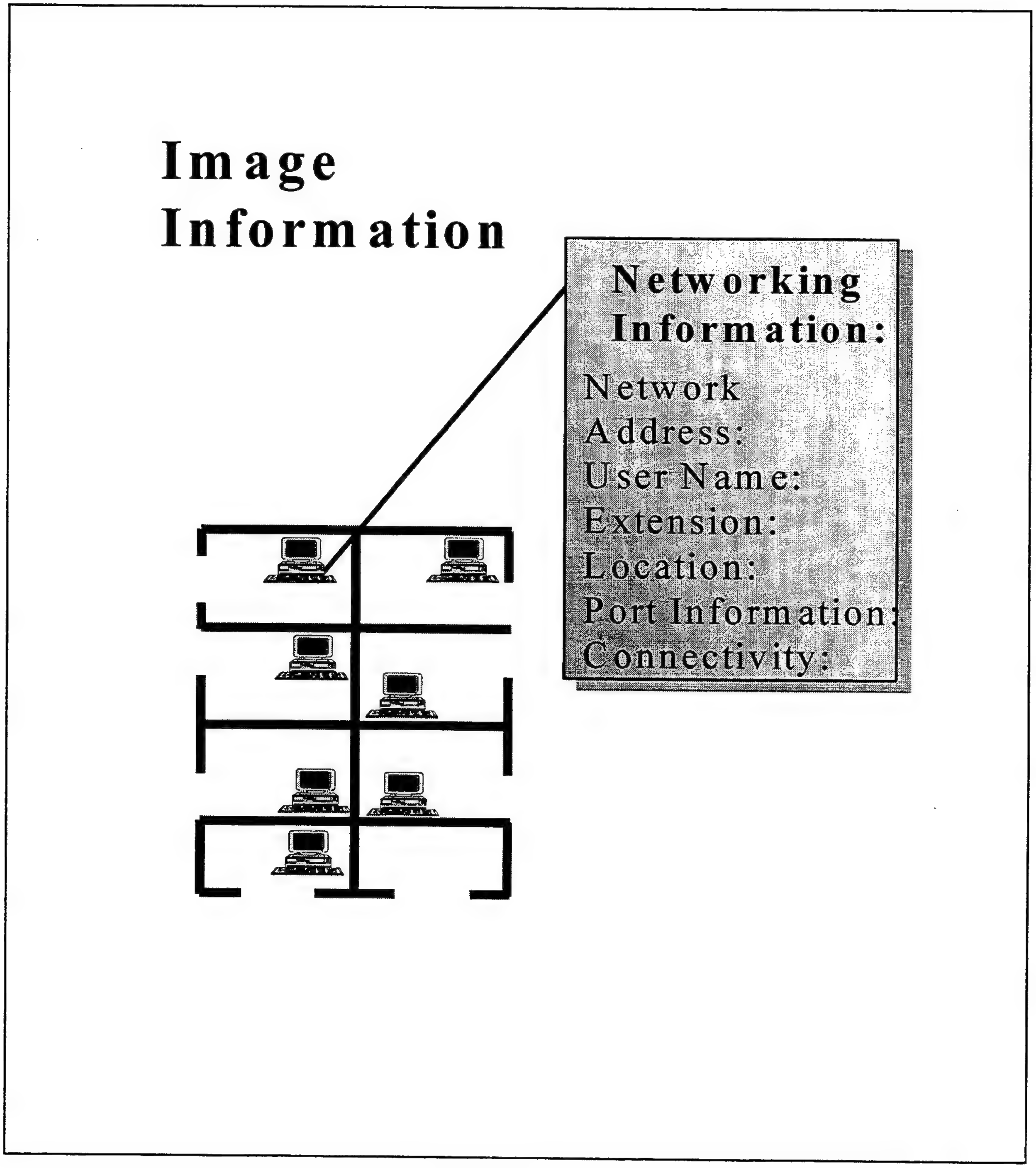


Figure 3-8

Results

The implementation of ClickNet Version 2.0 for analysis and management of the OIS unclassified network creates an easier method of managing this network. Although the program was not completely finished due to time restraints, the work accomplished will allow completion of the program to be made with little effort. In addition, many of the advantages and disadvantages of the software package have already been distinguished.

Disadvantages

ClickNet requires an excessive amount of memory to be used. The creation of a multiple layers of graphic images and backgrounds that are connected to a large network supplies database resulted in a very large file. Other problems arose in the connection of network images to segments to provide connectivity information. Finally, disadvantages were found in the limited amount of network images found in the software package and the limited ability to import images and backgrounds into the program.

Advantages

The graphical representation creates an easier format for the user of the program, allowing the user to view connectivity and other network information at the same time. The graphics environment also allows the user of the program to quickly and easily operate and view the program with only the use of a mouse. This environment also allows the user to easily make addtions and changes in network supplies and connectivity. The use of floorplans as backgrounds enable the user to see exactly where the device is located and how it is connected to the network. Finally, the diagrammed levels of network linked to an equipment database allow the user to query needed information within the graphics environment.

Conclusions

The implementation of ClickNet, an analysis and management program for the OIS (Office Information Systems) unclassified ethernet network, produced a more effective way of managing the network. However, there are many advantages and disadvantages of the use of ClickNet for use in this capacity at Arnold Engineering Development Center. The amount of memory required to run a ClickNet program for AEDC is large: however, this inconvenience has been overcome. The other problems including the importation of images and the ability to show connectivity of every image in the database have been discussed with representatives of PinPoint Software Corporation, the manufacturers of ClickNet Version 2.0. In conversations with these representatives it has been suggested that these problems may be solved in upcoming versions of the software package. It has been determined that the use of this software package would be an effective method of analysis and management of networks at Arnold Engineering Development Center; however, the difficulties that arose in the use of this software are still being evaluated to determine if it would be best to implement ClickNet for network management and analysis at AEDC.

DEVELOPMENT OF AN INVESTMENT PROJECT DATABASE

Derek E. Geeting

Shelbyville Central High School
401 Eagle Blvd.
Shelbyville, TN 37160

Final Report for:
High School Apprentice Program
Arnold Engineering Development Center

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

and

Arnold Engineering Development Center

August 1995

DEVELOPMENT OF AN INVESTMENT PROJECT DATABASE

Derek E. Geeting
Shelbyville Central High School

Abstract

A database to store and maintain information about investment projects for the current fiscal year was created. Database programming was first learned. Then, a working database was created. Next, data for the upcoming fiscal year (1996) was entered into the database. From this point data can be easily modified or rearranged. Also, informative reports can be created which give detailed information about the investment project data such as summaries of project data. Finally, the database was made expandable, in case the projects of other subtasks or sections need to be added later.

DEVELOPMENT OF AN INVESTMENT PROJECT DATABASE

Derek E. Geeting

Each fiscal year, a new contract is agreed upon between the Arnold Engineering Development Center (AEDC) support contractors and the Air Force. This contract is a detailed list of the goals that AEDC would like to accomplish each year for the Air Force. In order to reach an agreement on which projects should be carried out, how they should be carried out, and what allocations of funds should be made, each section at AEDC must negotiate with the Air Force until an agreement is made. Sometimes this can become a difficult task because of the large amounts of data that can change constantly. In order to first agree on a contract and then give the Air Force an accurate picture of where each project stands at various intervals throughout the year, an organized manner of data storage, maintenance, and retrieval was needed. A database with limited capabilities and a lack of centralization of materials made these tasks quite tedious. A new database that could easily store all investment project information and be maintained with little effort needed to be created.

The first step in developing an investment project database is to choose the software to handle the data. Microsoft Access is one of the most powerful tools on the market for PC database maintenance. It has almost unlimited capabilities and is very user-friendly. Also, it is the standard PC database tool for the Air Force.

After learning the Microsoft Access software package and its uses, the first step in creating the database for FY 1996 is to set up tables. Tables are the primary element of a database. Generally, they look and function much like a spreadsheet. However, MS Access is a relational database, which allows the user to relate data together from several tables. This provides an environment that is much easier to read and edit than a spreadsheet. A relational database allows each record in a table to be linked through the use of a primary key to one or several records in another table or tables. With multiple tables one can locate and edit individual sections of data with ease.

After setting up tables, the next step is to design queries. Queries function much like tables, but they are restrictive in the data that they contain. Queries allow you to search through all the records in a database and extract only those needed for a specific task. Mostly, queries are used to find specific data quickly or to define the records and sorting order of a report. In this project, queries were used mainly to set up data and organize it as it should be displayed in reports.

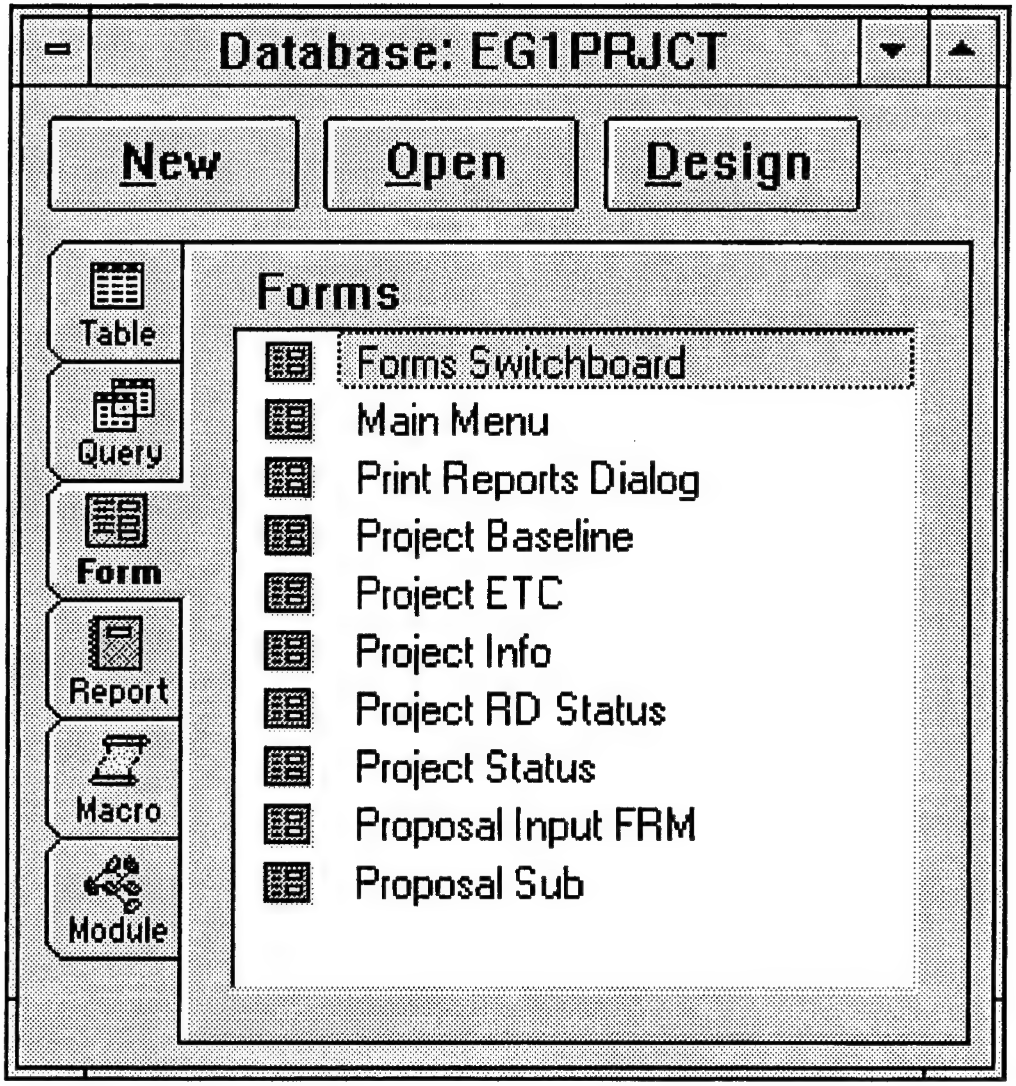
Another step toward completing the database project is to design forms. Forms are simply a means of entering data into tables in a quick and easy manner. They allow the user to enter information concerning a particular record in many tables at the same time. This allows little room for error in setting up proper links between tables and ensures maximum efficiency. Also, forms can be presented in a user-friendly, graphical environment that provides for database maintenance by those with the least amount of experience with database knowledge. Forms are the most essential element of the database as far as interaction between the file and the user is concerned.

Reports are also a very important element of databases. The reports are actually the end product for which the database is originally created. Reports simply give the user a detailed description of the data in the tables in an organized, printed form. This allows the database user to make application to the data through analyzing the data and patterns therein. Also, the reports can offer summaries in relation to the data entered. While forms are the essential element of input in a database, reports are the essential element of output.

Another tool implemented by the database is macros. Macros are not a pertinent part of the database, but they are probably the most powerful. A macro simply automates certain processes. This allows a database programmer to design the database to be very user-friendly and extremely efficient in time and error management. For example, after defining a report one may need to go through at least five or six more steps before a printed copy is in hand. A macro can reduce this task to simply clicking on a button in a dialog box. Macros hold the future to the investment project database for section EG1. As other systems become more easily accessible, the database will be able to be linked to other sources and instantly download data instead of having to key it in by hand. To some degree this is possible now, but because of the current status of other data sources, this is not quite the most efficient process. Macros, however, were implemented with great success into this database project.

The EG1 investment project database is well on its way to completion. It will serve as a very useful tool to this section, and hopefully it can be used efficiently by others as well. Enough construction has been completed on the database that it was able to handle the data for the upcoming contract for FY 1996. It was used in maintaining the data, such as sorting and allowing for changes to the projects as new information was received. A contract report was set up and is now being used to provide contract information. The investment project database has undergone much structural development and will continue to be refined in use at AEDC.

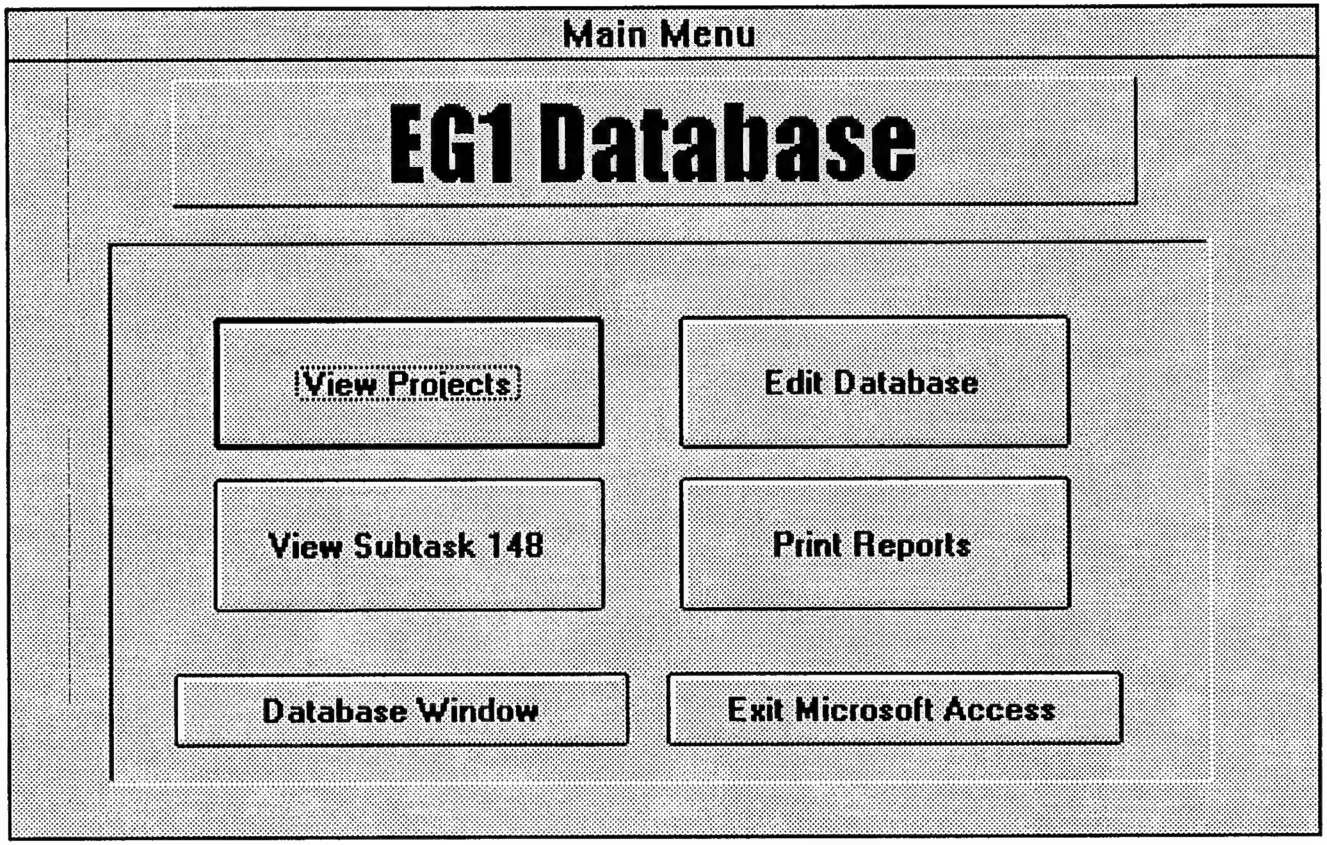
Database Window

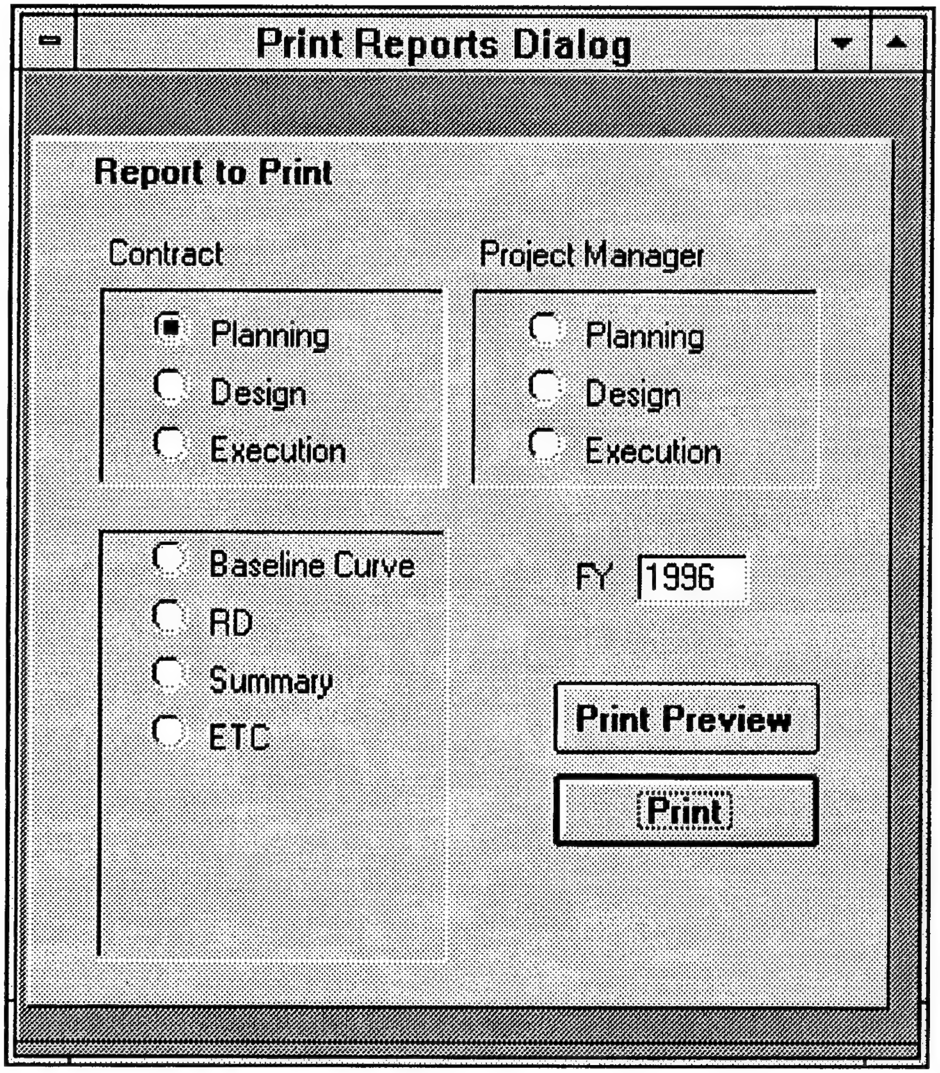


Forms

- Forms Switchboard
- Main Menu
- Print Reports Dialog
- Project Baseline
- Project ETC
- Project Info
- Project RD Status
- Project Status
- Proposal Input FRM
- Proposal Sub

Main Menu Form





Print Reports Dialog

Report to Print

Contract

● Planning
○ Design
○ Execution

Project Manager

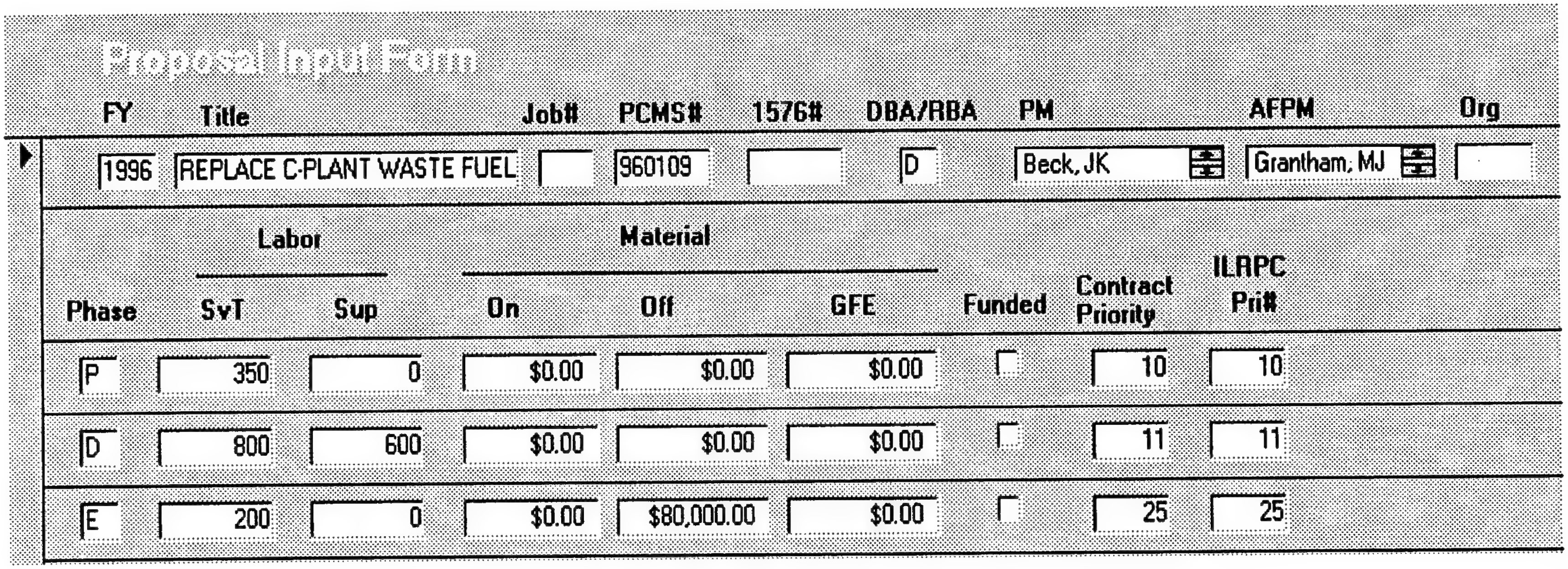
○ Planning
○ Design
○ Execution

○ Baseline Curve
○ RD
○ Summary
○ ETC

FY 1996

Print Reports Dialog

Project Input Form



Proposal Input Form

FY	Title	Job#	PCMS#	1576#	DBA/RBA	PM	AFPM	Org
1996	REPLACE C-PLANT WASTE FUEL		960109		D	Beck, JK	Grantham, MJ	

	Labor		Material					
Phase	SvT	Sup	On	Off	GFE	Funded	Contract Priority	ILRPC Pri#
P	350	0	$0.00	$0.00	$0.00	□	10	10
D	800	600	$0.00	$0.00	$0.00	□	11	11
E	200	0	$0.00	$80,000.00	$0.00	□	25	25

			SvT		Sup		On	Off			Cumul.
Job	**Pri#**	**Title**	**MHR**	**MHR$**	**MHR**	**MHR$**	**Contract**	**Contract**	**GFE**	**Total**	**Total$**
	1	REPL VA-1309 COMPR ROTORS,V	300	$10,227	0	$0	$0	$0	$0	$10,227	$10,227
	2	REPL CENTRIFUGAL COMP ROTORS	300	$10,227	0	$0	$0	$0	$0	$10,227	$20,454
	3	FAB/REPL JACKSHAFT PWT-MD	600	$20,454	0	$0	$0	$0	$0	$20,454	$40,908
	4	REPL 16 T/S MD MTR BRG SEALS	1,500	$51,135	0	$0	$0	$0	$0	$51,135	$92,043
	5	REPL PWT DRIER BURNER CONTROLS	2,800	$95,452	100	$2,648	$0	$56	$0	$98,156	$190,199
	6	REFURB PWT DRIER BEDS	960	$32,726	60	$1,589	$3,000	$0	$0	$37,315	$227,514
	7	REPR VACCUM LEAKS-LN2 VAC JAC	500	$17,045	0	$0	$0	$0	$0	$17,045	$244,559
	9	ADD REFERIG PUMPOUT UNITS B&C PLANT	3,186	$108,611	20	$530	$0	$111	$0	$109,251	$353,811
0619	10	REFURBISHING A-PLT FUEL SYSTEM	6,380	$217,494	120	$3,178	$2,000	$500	$0	$223,172	$576,982
	11	REPLACE C-PLANT WASTE FUEL TANK	800	$27,272	600	$15,888	$0	$0	$0	$43,160	$620,142
	12	REPAIR A-PLANT PRESSURIZED COOLER	3,618	$123,338	192	$5,084	$1,500	$1,000	$260,000	$390,922	$1,011,064
	13	REPL FRICK FUEL REFER SYSTEM	700	$23,863	100	$2,648	$0	$0	$0	$26,511	$1,037,575
	14	SPARE STUB SHAFT A-PLT AIRSIDE COMPR	250	$8,523	0	$0	$0	$0	$0	$8,523	$1,046,098
2173	15	REPL ETF-A/B SIG COND PANELS	0	$0	0	$0	$0	$0	$0	$0	$1,046,098
2202	16	REPR B-PLT A3 COMP GERMAN VALVES	0	$0	0	$0	$0	$0	$0	$0	$1,046,098
		Totals:	**21,894**	**$746,366**	**1,192**	**$31,564**	**$6,500**	**$1,667**	**$260,000**		**$1,046,098**

ARCHIVING THE TEST FACILITIES HANDBOOK USING TPAS

Andrew Kaczorek

Tullahoma High School
1001 North Jackson Street
Tullahoma, TN 37388

Final Report for:
High School Apprentice Program
Arnold Engineering Development Center

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

and

Arnold Engineering Development Center

August 4, 1995

ARCHIVING THE TEST FACILITIES HANDBOOK USING TPAS

Andrew Kaczorek

Abstract

The Test Facilities Handbook has been used for many years to orient new customers to Arnold Engineering Development Center. The purpose of this project was to assist of the conversion of the Test Facilities Handbook into HTML form. This allowed the documents to be viewed by any platform supporting NCSA Mosaic. It also allowed documents to be both searchable and intuitively linked together by hyperlinks. Data that were received in raw form, such as photos, tables, schematics, and video, were manipulated into appropriate formats and included in the final version. The result of this project is an electronic version of the Test Facilities handbook that is modern, interactive, and compact.

ARCHIVING THE TEST FACILITIES HANDBOOK USING TPAS

Andrew Kaczorek

General Description of Research: What, Why, Results, Applications to Larger Research Effort

AEDC has used the Test Facilities Handbook to introduce prospective customers to the facilities and types of testing available. However, as corporations became more computerized, it became necessary to create an electronic version for reference purposes. When authoring electronic reference material, it is imperative to allow all platforms to support it. At this point in time, the most universal language is Hypertext Markup Language (HTML). Internet browsers such as Mosaic and Netscape can read HTML and display it as text, pictures, and hyperlinks. Hyperlinks are a major advantage of HTML. They allow a hierarchical system of files to be linked together by clicking on linked phrases which appear a different color than the surrounding text. This allows for an interactive display of text, diagrams, and even digital video. Information is easier to find because related documents are linked together and a search can be performed of the entire text for keywords. The entire system, including the browser, viewers, text, and multimedia, can be written on to one compact disk. This method of archival, the Test Project Archival System (TPAS), has proven to be a reliable, compact, and superior form of documenting the facilities at Arnold Engineering Development Center.

Detailed Description of Research: Methodology (process, approach, etc.), Apparatus/Equipment used, Analysis Techniques

The Test Facilities Handbook is a publication that is produced for the purpose of orienting prospective customers to the facilities, services, and expertise available at Arnold Engineering Development Center. It gives detailed descriptions of the test cells, types of testing, systems tested, and modifications available to meet testing needs. It's complete listing of information about AEDC also makes it an excellent reference tool for engineers and technicians who need facts about systems or procedures that they are unfamiliar with. The Test Facilities Handbook provides a complete listing of the policies and procedures governing the planning and conduct of testing and evaluation at AEDC.

Although the current version of the handbook is adequate, its in-depth descriptions and large amount of data included make it large and saturated with information. Certain facts are difficult to find due to the large volume of details and available figures. As the information age becomes more pronounced in the engineering marketplace, the need arises to produce a modern, computerized version of the Test Facilities Handbook. However, one difficulty in producing computerized software is to provide equal compatibility between all platforms that might be available to prospective customers.

With the creation of the Internet, this problem was also addressed. Users all across the world own numerous types of systems. To make a worldwide computer network, a standard interface and language needed to be devised. This language is HTML, Hypertext Markup Language. HTML allows users on many varieties of computer types to access the same information and have it displayed in the same manner. It provides a standard form for text, formatting, and directory structure. Files within HTML documents can be linked together by hypertext, highlighted text that, when selected, loads another document pertaining to the highlighted text.

To view an HTML file system, an HTML browser such as NCSA Mosaic and Netscape are used to display and interact with the documents. These programs are free of charge and can be downloaded from many sites on the Internet. NCSA Mosaic was chosen for distribution purposes for the handbook. For the Test Although the Test Facilities Handbook is not running on the Internet, the system works in the same manner. All files are arranged logically into directories for use in menus. Related documents are intuitively linked together by hypertext, making the entire handbook interactive.

In order to create hyperlinked documents, all text files must be created in HTML format. Special formatting and styling codes are needed to fully take advantage of the graphical browsers' capabilities. These codes must be added directly by a programmer, or automatically added via a HTML editor. HTML is a language that can be written in a simple text editor such as Windows Notepad. This is the purest form of authoring. HTML commands are directly inputted and saved in a text file. All images and hypertext are programmed into the file where they should appear. Simple editing tools are available, but extensive editing becomes complex during fully text programming.

However this method of authoring is cumbersome and can be simplified using several helpful tools. One tool for HTML authoring is the Internet Assistant add-on for Word 6.0. Within the Internet Assistant, one is able to see HTML as it should appear in the browser displaying it. Images that are inlined by HTML commands are actually inlined on the screen. This display makes it easier for an HTML programmer to envision the finished product. Also, more complex editing features are available to the user. Internet Assistant greatly simplifies HTML programming, but in a complex HTML system, the links between a great number of files can become overwhelming.

To simplify the file structure, a program was written that organized HTML files in directories. The program, Tpastool, would first build a list of each file in every directory. Then, in each directory, a master.lst file was created that contained a list of every file and subdirectory within that directory. After the list was built, Tpastool then built an HTML file in every directory corresponding to the name of the directory. These HTML files gave links to every file and subdirectory with hypertext of the name of the file or directory. The programmer could edit the master.lst to change the text of each hyperlink to fully explain the link.

Other files were acquired to enhance the quality of the finished product. Line art, schematics, photographs, and real time video were linked into the HTML. High-resolution scanners were used to digitize the schematics and photographs. Much manipulation and conversion was required to achieve a quality premium enough to be included in the final version of the Test Facility Handbook. A video capture board was used to digitize the video of testing at AEDC. Several types of video compression were used to make the videos suitable for the data rate of a CD-ROM. All multimedia items were sized and compressed in a way that would make them compatible and viewable on all platforms supported by HTML.

The equipment used in creating the HTML Test Facilities Handbook is as follows: Pentium-90 PC, Video Capture Board, Yamaha CDR compact disc recorder, HP DeskJet 1200c printer, Canon BJ-600 printer, Panasonic video cassette recorder, and Panasonic television monitor. Software used in it's development is as follows: NCSA Mosaic, Microsoft Word with Internet Assistant, Corel Draw 3 and 4, Micrografx Picture Publisher 5, Adobe Premiere 1.1, Microsoft Video 1.1, Asymetrix Digital Video Producer, TPAS Toolkit, Hijack Professional 3, and Lview 3.1.

Throughout the production of the Test Facilities Handbook, advances made were organized into versions known as alpha versions. These versions not only allowed to chart the progress of the handbook, but to revert back to previous material in the case of a data loss or mistake. The handbook progressed in many stages; the three major sections of the handbook, ETF, PWT, and VKF were milestones in its creation. Further ongoing refinement made the handbook more intuitive and graphically pleasant. The handbook will continue to change as AEDC changes. Also, corrections will be made in the future that will be helped by the electronic handbook's comments section.

Another advantage of this electronic version is electronic feedback. Users may ask questions or comment about the handbook or its contents by three methods, standard mail, email, and internet http mail. Prospective customers can inquire about more information on specific topics found in the book. Also, they may request larger or more detailed multimedia items included on the CD-ROM by the numbering system, TFH, used in labeling items in the handbook. Tables, images, and text may be sent to those without a computer; plans are underway to create an updated hard copy of the electronic Test Facilities Handbook.

The TPAS CD-ROM archival of all types of test procedures and data, like the Test Facilities Handbook, is a fantastic breakthrough in technology. Just its usability makes it far superior to any method available in the past. An even greater advantage of TPAS is the size of a CD-ROM. Test projects that would have taken an entire shelf of books, data, and statistics can fit into the palm of a hand. Shipping of CD-ROMs is negligible to every corner of the world. Eventually, the HTML of TPAS may be fully taken advantage of if test data were available on the internet. Although this is possible now, the sensitivity of documents contained on these CD-ROMs currently make it unwise to network it. However, this version of the handbook has countless future possibilities and advantages.

The result of the complete HTML Test Facilities Handbook is a far superior form or archival. Users may search the entire database for an individual word or phrase. It is not necessary to thumb through numerous pages to find the section that one is looking for. Additionally, the overall quality and cosmetic appearance of the handbook is far greater. Although the quality of the handbook should not be the deciding factor in a prospective customer, it is subconsciously. The quality that the handbook impresses upon the user will undoubtedly reflect the quality of the facilities and personnel at Arnold Engineering Development Center. This major step in modernizing the handbook should prove to be a worthwhile project by an increased understanding and respect for AEDC by every user interested in testing their product.

Results: What was observed or discovered and conclusion

As new ways are being discovered of storing and retrieving data, many new methods of archival will surface. At this point in time, the most widespread and multi-compatible language is HTML. Although the language may change in the future, the interactivity of hypertext should continue to be a trend in the information age. This new form of the Test Facilities Handbook is superior to the paper version and should be used for many years in the future.

Acknowledgments: Thanks to your mentor(s), others

I would like to thank Steve Lodholz, my mentor for his assistance in my learning of HTML through practical and useful applications. I would also like to thank Elizabeth Bricken, my coworker, who also assisted me with TPAS. Both of these individuals were a great help in my brief but enjoyable employment at Arnold Engineering Development Center.

ROCKET MOTOR BALLISTIC MODELING

David R. Landry

Franklin County High School
925 Dinah Shore Boulevard
Winchester, TN 37398

Final Report for:
High School Apprentice Program
Arnold Engineering and Development Center

Sponsored by:
Air Force Office of Scientific Research
Boiling Air Force Base, DC

and

Arnold Air Force Base

August 1995

ROCKET MOTOR BALLISTIC MODELING

David R. Landry
High School Apprentice
Franklin County High School

Abstract

The objective of my project was to evaluate the new SBRAM code for rocket motor ballistic modeling. This evaluation would offer a comparison of the final code to the developmental code. With this objective, I employed the SBRAM code with the three different rocket motor finite element models. Then, using motor chamber pressure as a guideline, I trimmed the models by modifying input parameters in the computer ballistic RECESS code. The untrimmed models revealed the errors in the ballistic module/code inputs. However, trimming the models provided insight into the sensitivity of the input parameters, useful for future revisions to the rocket motor models.

ROCKET MOTOR BALLISTIC MODELING

David R. Landry
High School Apprentice
Franklin County High School

Introduction

Background Information

SBRAM, which is an acronym for Structural Ballistic Risk Assessment Methodology, was developed by Thiokol in a joint effort with Philips Lab, Edwards Air Force Base, CA., and Arnold Engineering and Development Center. The function of the SBRAM code is to predict the structural and ballistic performance of flawed solid rocket motors. SBRAM is a broad term; it actually covers various codes. FEINT, RECESS, Supertab, and Matter were the particular codes utilized in my project. Obviously, by looking at Figure 1, these particular codes were selected simply because these are the ones offered at AEDC. These codes are all executed from FEINT.

Discussion of Problem

Recently, Thiokol developed the final version of the SBRAM code. It differs from the earlier developmental code because it operates on a workstation instead of a mainframe, and additional features were implemented in the SBRAM code. However, the same finite element models were exercised as were the inputs to the ballistic code RECESS. In the past, engineers operated the developmental version of the SBRAM code. Now, the variations between the codes needed to be assessed, and the inputs to the ballistic RECESS code needed to be evaluated for sensitivity and trimmed to specific rocket motor conditions. With this goal, I have used prior work and combined it with my work to provide a recommendation for future revisions to rocket motor models.

Methodology

The first few steps of my project developed my understanding of the new SBRAM code. I discussed the code with engineers, read user's manuals for the code, reviewed work previously accomplished, and experimented with the new SBRAM code. Then, once I was accustomed to the code, I employed the SBRAM code using the various structural finite element models (Figure 2) for the Peacekeeper Stage II, Minuteman Stage II, and the Minuteman Stage III rocket motors. First, I noticed there were differences in the motor chamber pressure for the Minuteman Stage III in the developmental and the final SBRAM code. However, there were insignificant differences in the chamber pressure for the Minuteman Stage II and the Peacekeeper Stage II in the old and new code. Realizing that the Minuteman Stage III was atypical, I decided to employ the Peacekeeper Stage II as part of my next evaluation.

In my next evaluation, I attempted to trim the Peacekeeper Stage II model to correspond to the PQA-4 test data. Using motor chamber pressure as a guideline (Figure 3), I modified the propellant density, nozzle throat area, motor chamber volume, and port area in the input deck of the model. Eventually, a final trimmed model was constructed.

Results

The final code compared favorably with earlier versions of the Peacekeeper Stage II and Minuteman Stage II models but not with the Minuteman Stage III model. In Figures 4, 5, and 6, comparisons of the developmental and final SBRAM code for the various models are shown. There is infinitesimal variance between the codes, except for the Minuteman Stage III model.

When the final code for the Peacekeeper Stage II model was contrasted with the PQA-4 test data, the distinction is great (Figure 7). Trimming the model improved the agreement to measured motor test performance data, but additional trimming will be required. Each separate adjustment gave different results. Figures 8, 9, 10, and 11 reveal the results when the distinct properties were modified. Revising the propellant port area and the propellant volume had little effect. In comparison, modifying the density produced the best result. Although trimming the propellant throat area produced a distant pressure trace from the test data, the revision best simulated the trends of the data. Figure 12 discloses the final trimmed model. In the final trimmed model, the trimmed propellant throat area, volume, and density are applied. The final trimmed model explicates the result when those components were altered.

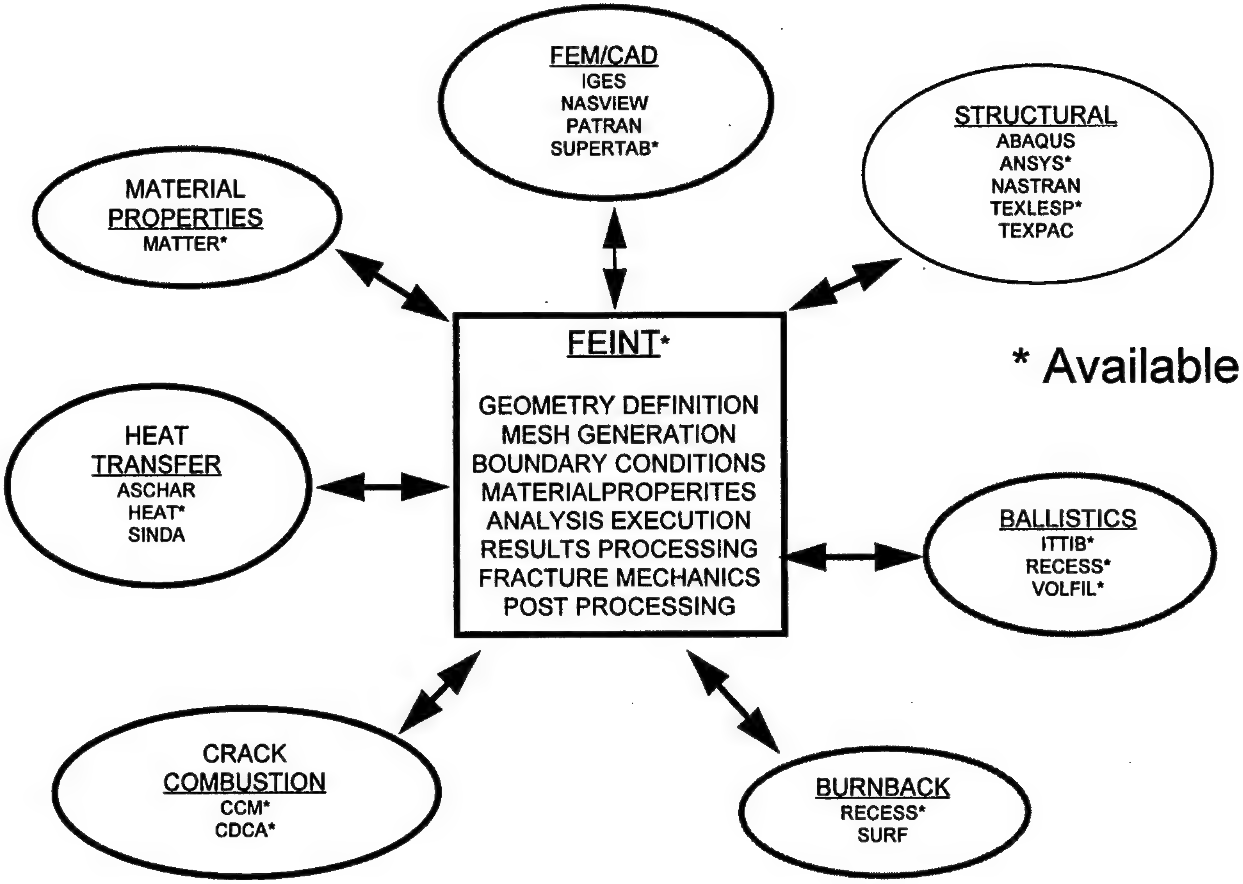
Benefits

I demonstrated the adequacy of the final SBRAM code. Obviously, the final SBRAM code requires some adjustment to its inputs before it can simulate test data. Next, I provided an initial trimmed model to the A & E Branch which is profitable in demonstrating the future revisions that need to be executed to the rocket motor models. Additional components to the rocket motor models should be altered to manufacture a similarity to the test data. Finally, I identified a potential problem with the Minutemant Stage III finite element model. The pressure trace for the Minuteman Stage III model was an oscillatory curve; the model requires a dominant smooth curve.

Acknowledgments

I would like to thank Max Roler for sponsoring me at Arnold Engineering and Development Center. He was extremely beneficial in satisfying my curiosity of rockets, providing guidance, and helping me settle in at AEDC. Also, I am grateful for Jon F. Seely for teaching me about the SBRAM code. I am extremely appreciative for his patience and understanding. To the other guys at the A & E Branch at AEDC, thank you for making me feel welcome and for answering all of my questions. Finally, thank you James D. Mitchell for sponsoring the high school apprenticeship program at AEDC. Mr. Mitchell is perfect for the program.

SBRAM: Structure Ballistic Risk Assessment Methodology Code



FEINT*

GEOMETRY DEFINITION
MESH GENERATION
BOUNDARY CONDITIONS
MATERIALPROPERITES
ANALYSIS EXECUTION
RESULTS PROCESSING
FRACTURE MECHANICS
POST PROCESSING

MATERIAL PROPERTIES
MATTER*

FEM/CAD
IGES
NASVIEW
PATRAN
SUPERTAB*

STRUCTURAL
ABAQUS
ANSYS*
NASTRAN
TEXLESP*
TEXPAC

HEAT TRANSFER
ASCHAR
HEAT*
SINDA

BALLISTICS
ITTIB*
RECESS*
VOLFIL*

CRACK COMBUSTION
CCM*
CDCA*

BURNBACK
RECESS*
SURF

* Available at AEDC

Figure 1. Various codes of SBRAM

FINITE ELEMENT MODELS: BURNBACK PROFILES

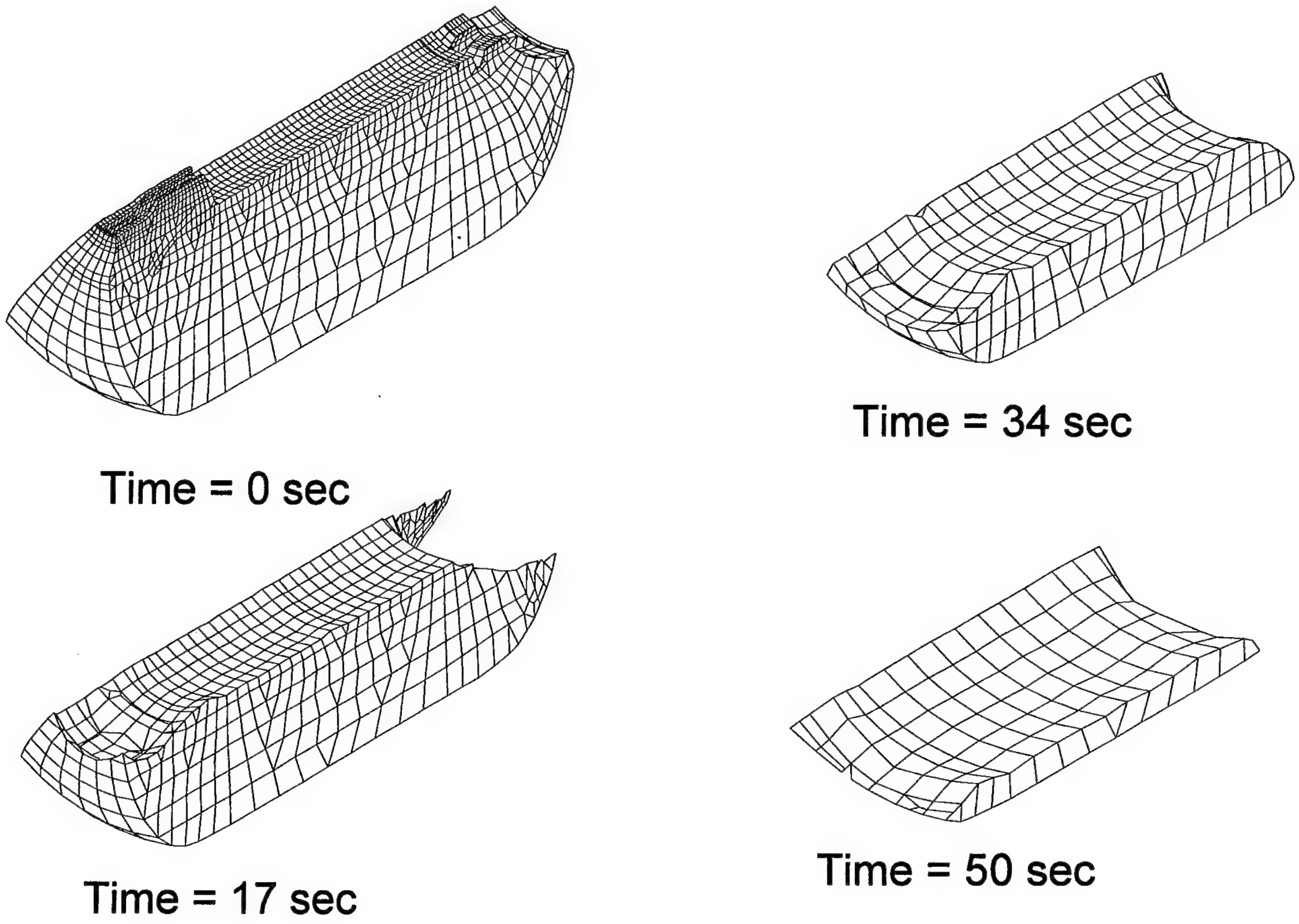
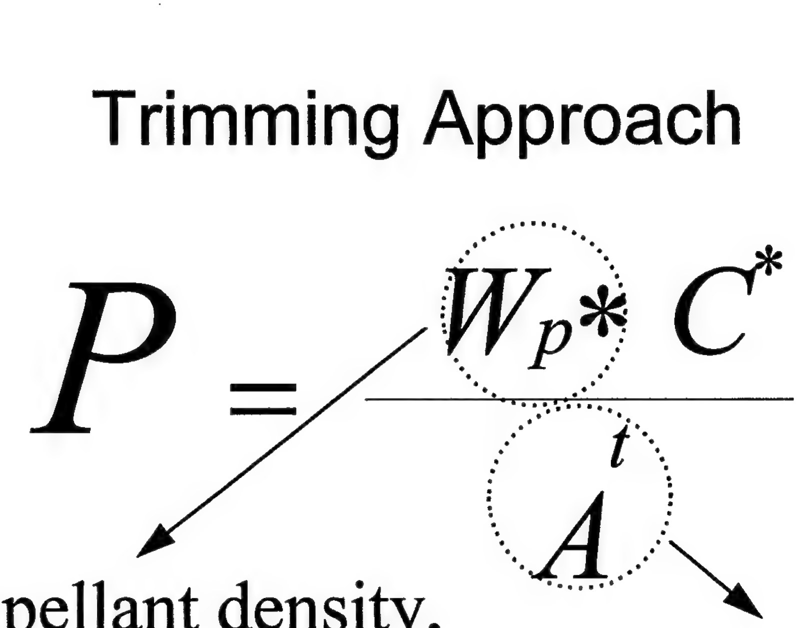


Figure 2. A sample of a burnback of a finite element model from 0 to 50 seconds

Trimming Approach



$$P = \frac{W_p * C^*}{\overset{t}{A}}$$

Affected by propellant density, and together with motor chamber volume and port areas combine to influence pressure

Throat Area is a function of motor burn time

W_p=Massflow Rate
C^*=Characteristic Velocity
A^t=Nozzle Throat Area

Figure 3. Formula for chamber pressure of a rocket motor

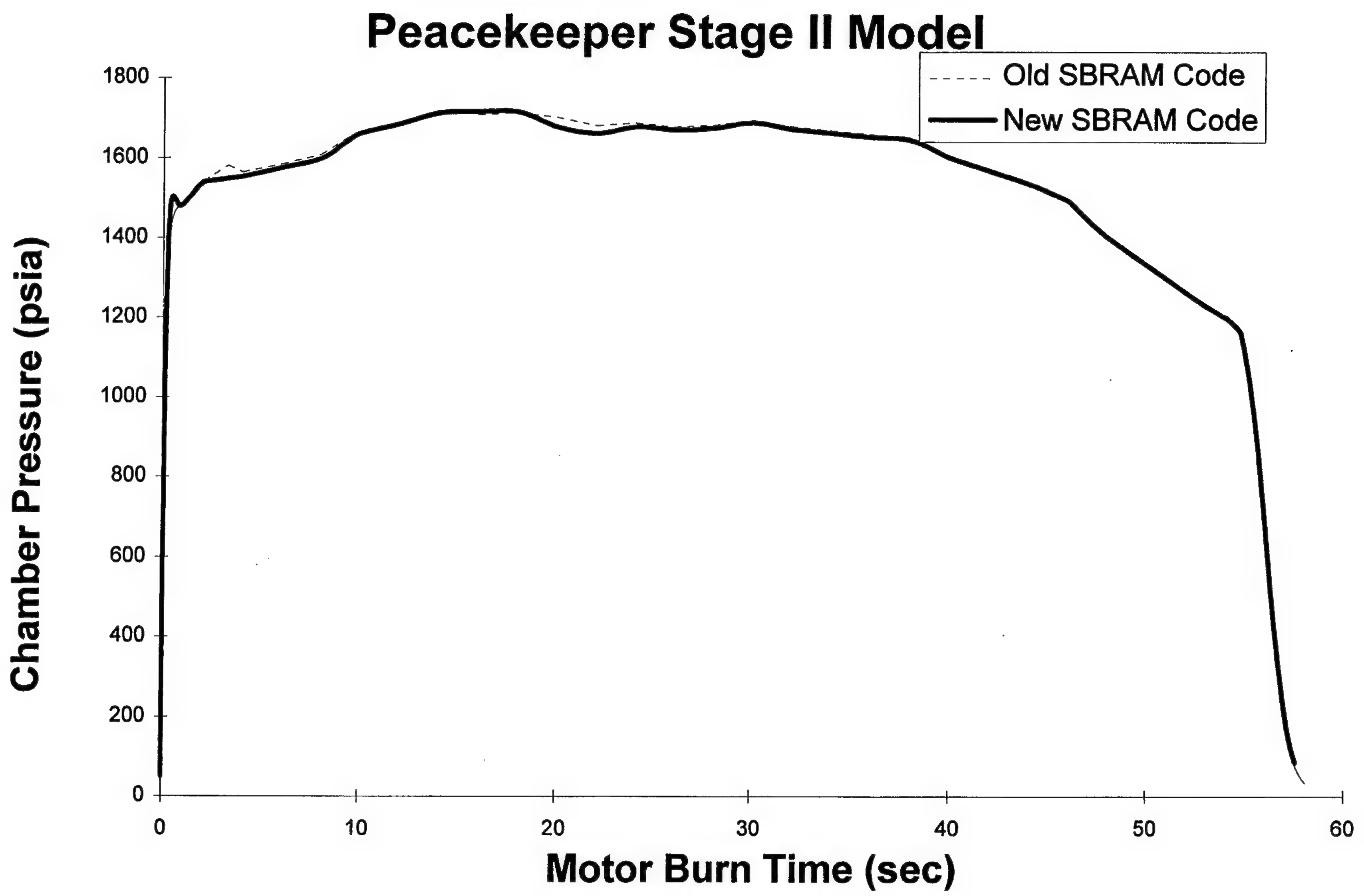


Figure 4. Comparison of the developmental and final SBRAM code for the Peacekeeper Stage II model.

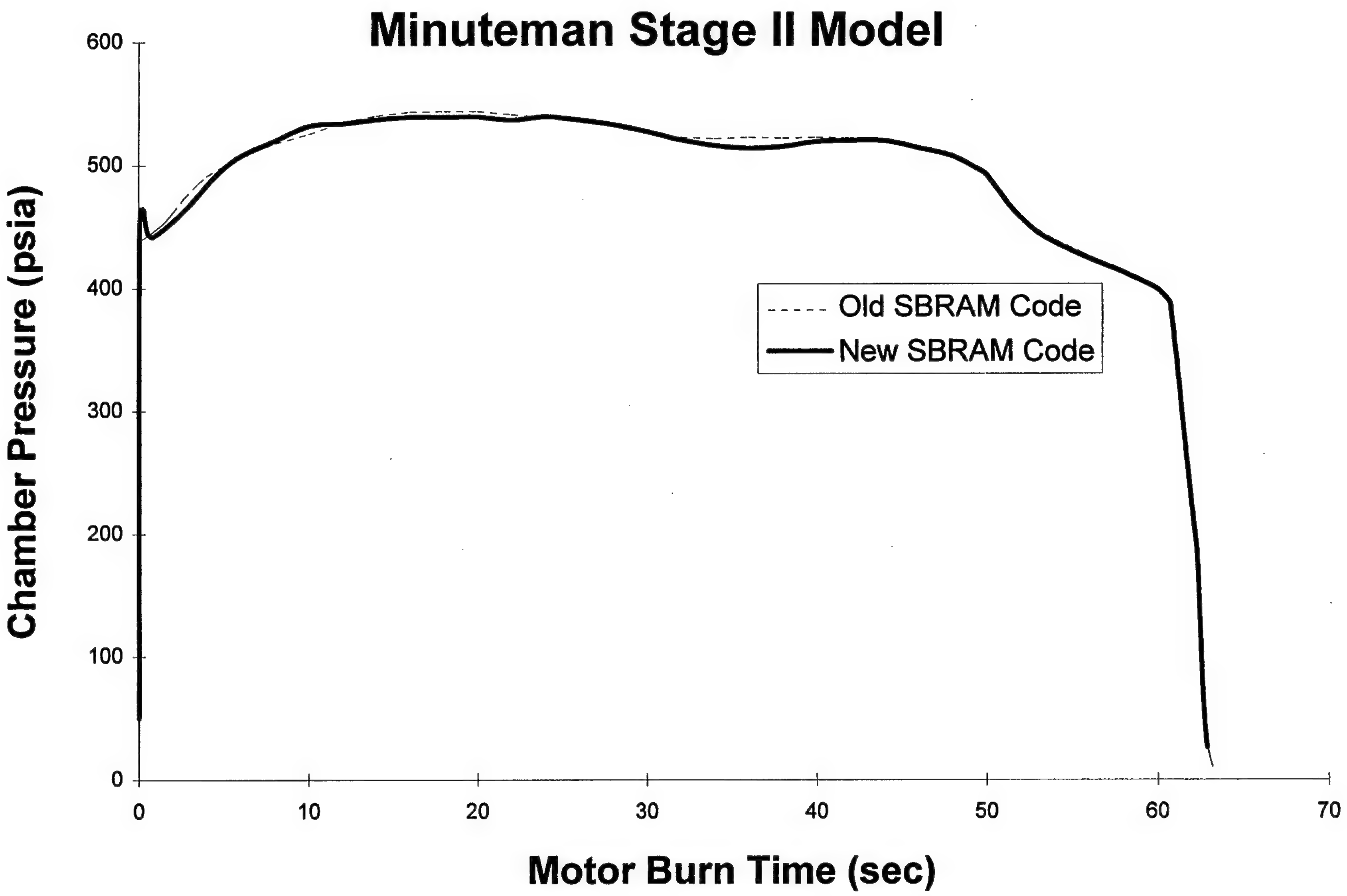


Figure 5. Comparison of the developmental and final SBRAM code for the Minuteman Stage II Model.

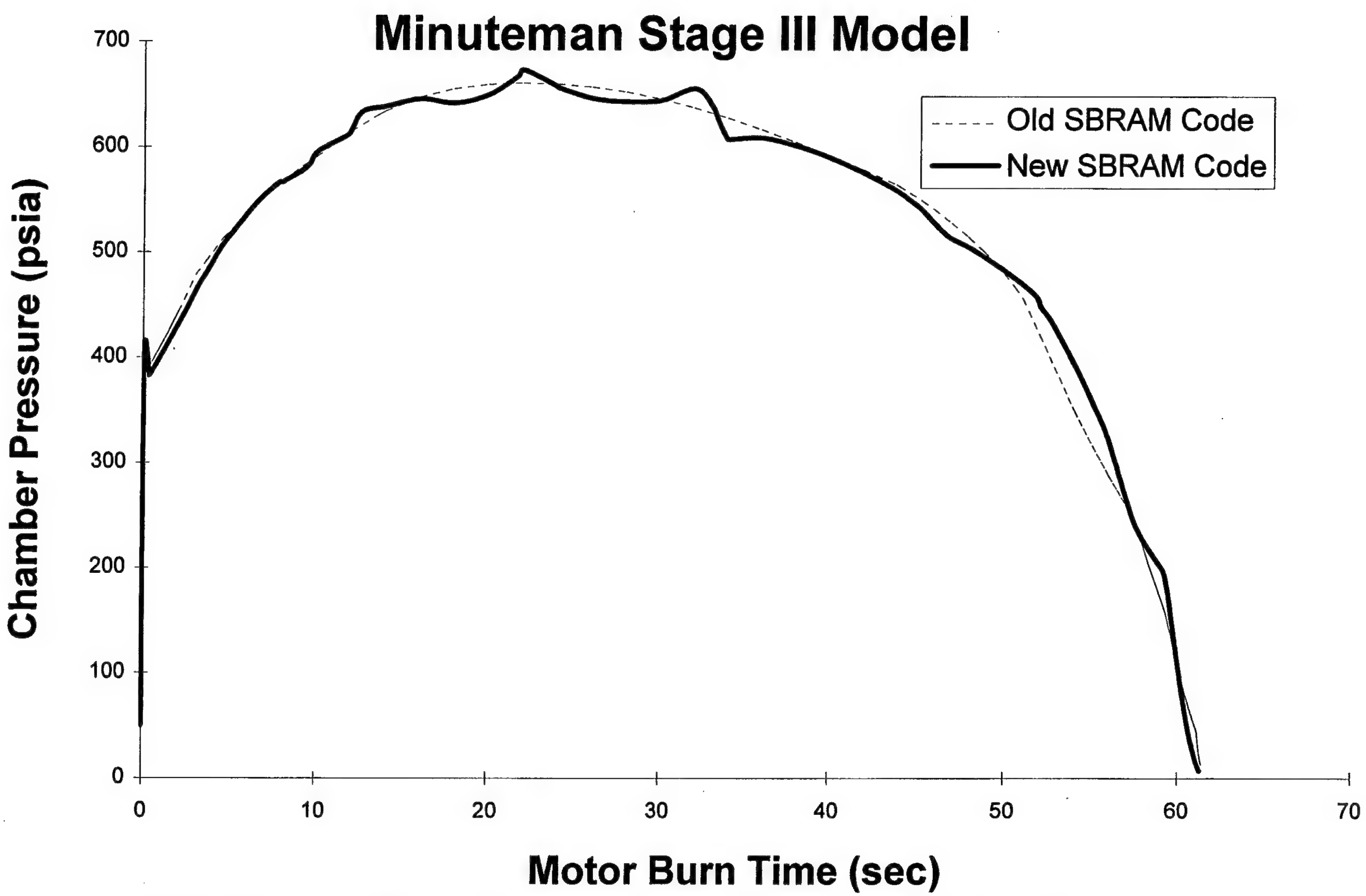


Figure 6. Comparison of the developmental and final SBRAM code for the Minuteman Stage III model.

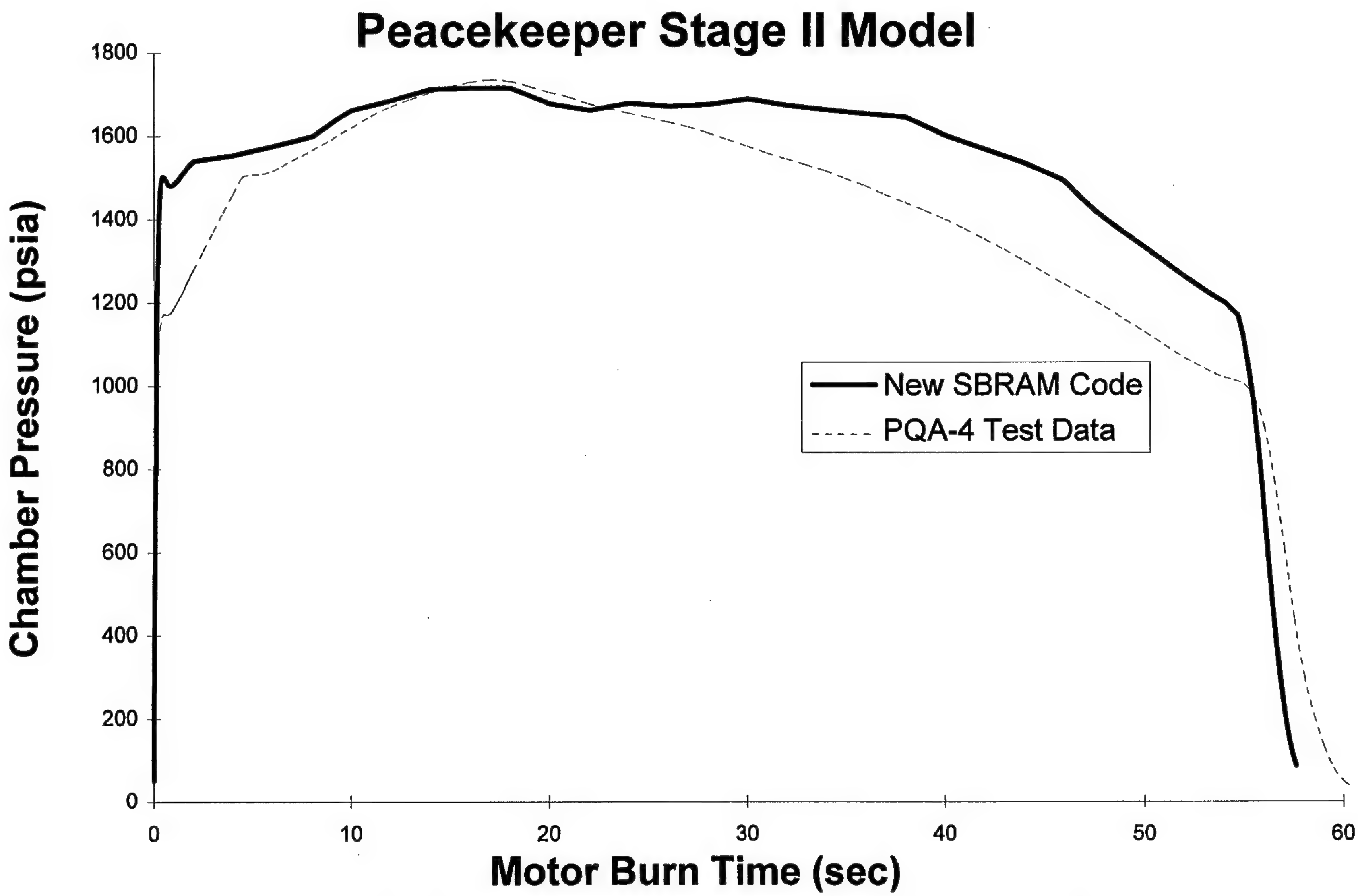


Figure 7. Comparison of the final SBRAM code and test performance data for the Peacekeeper Stage II model.

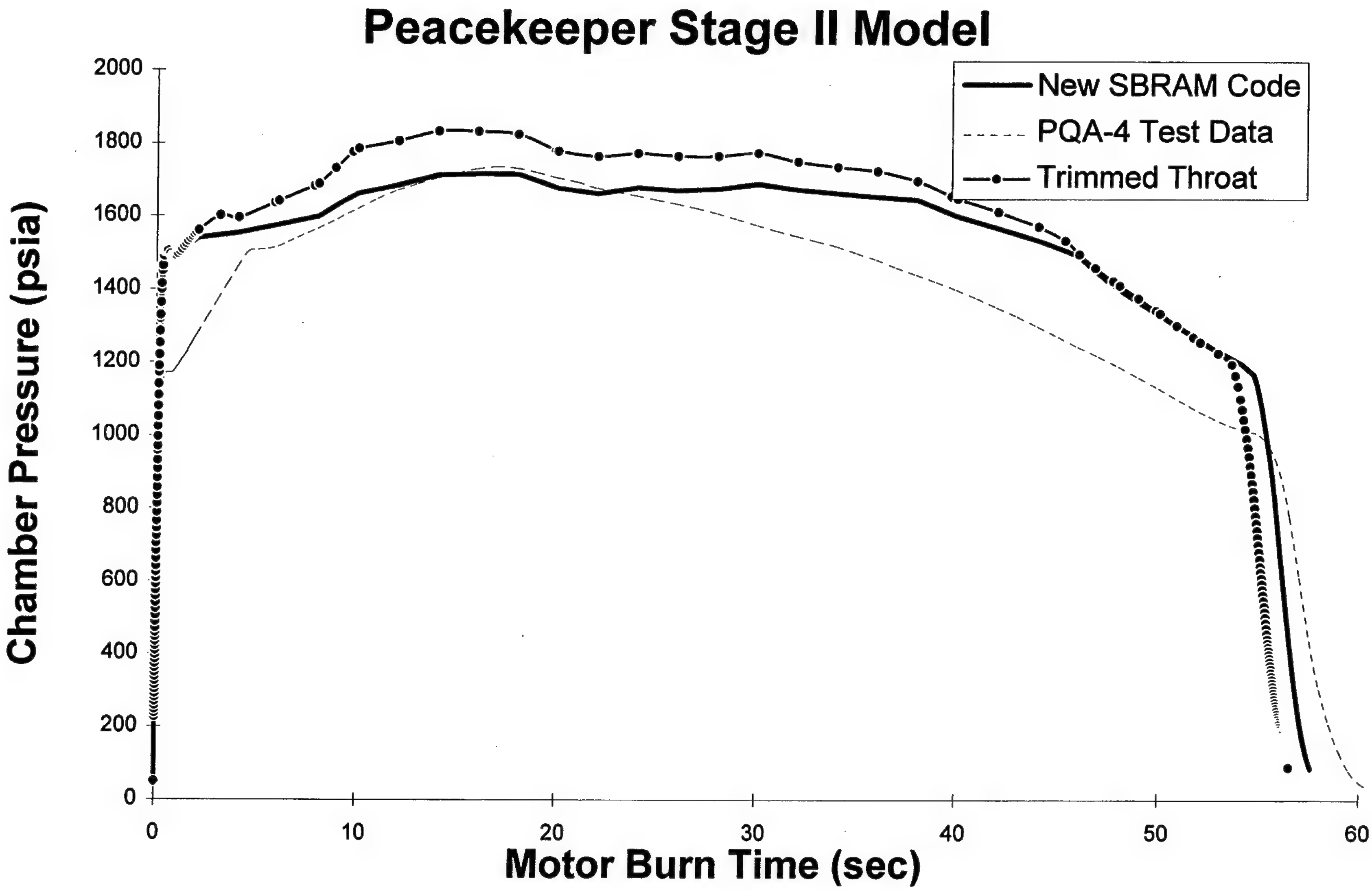


Figure 8. Comparison of the new code, test performance data, and trimmed throat area.

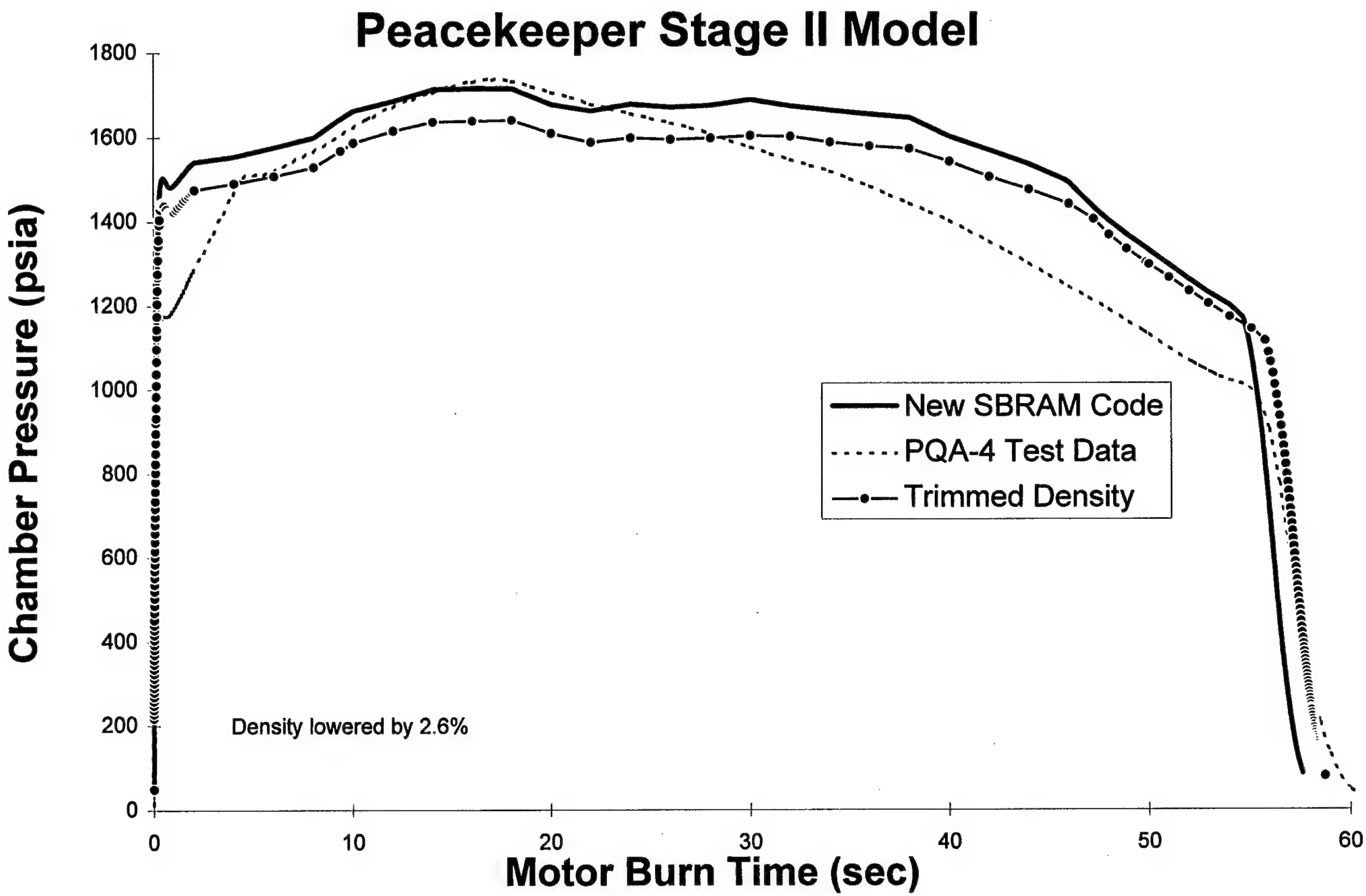


Figure 9. Comparison of the new code, test performance data, and trimmed propellant density.

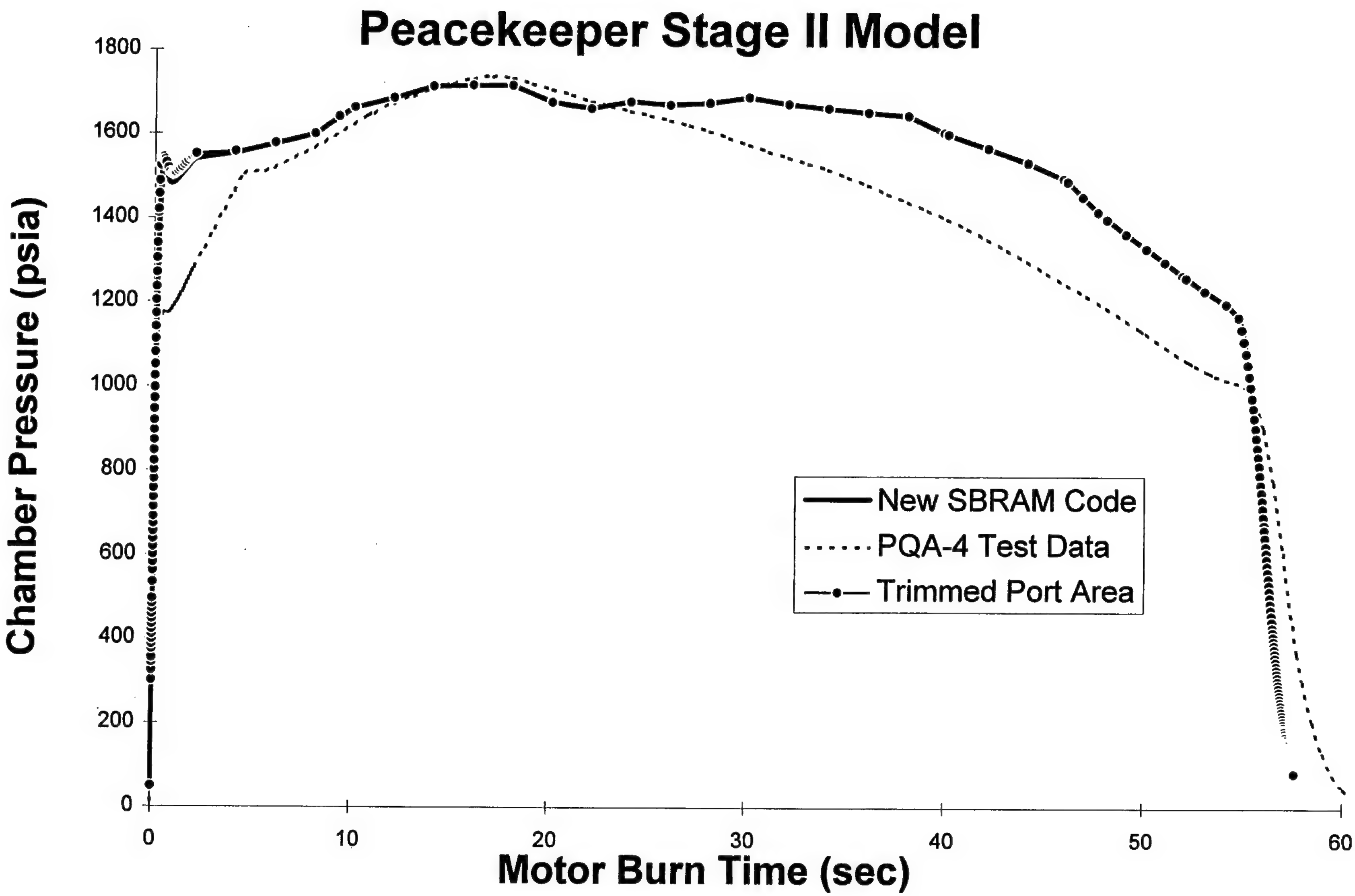


Figure 10. Comparison of the new code, test performance data, and trimmed propellant port area.

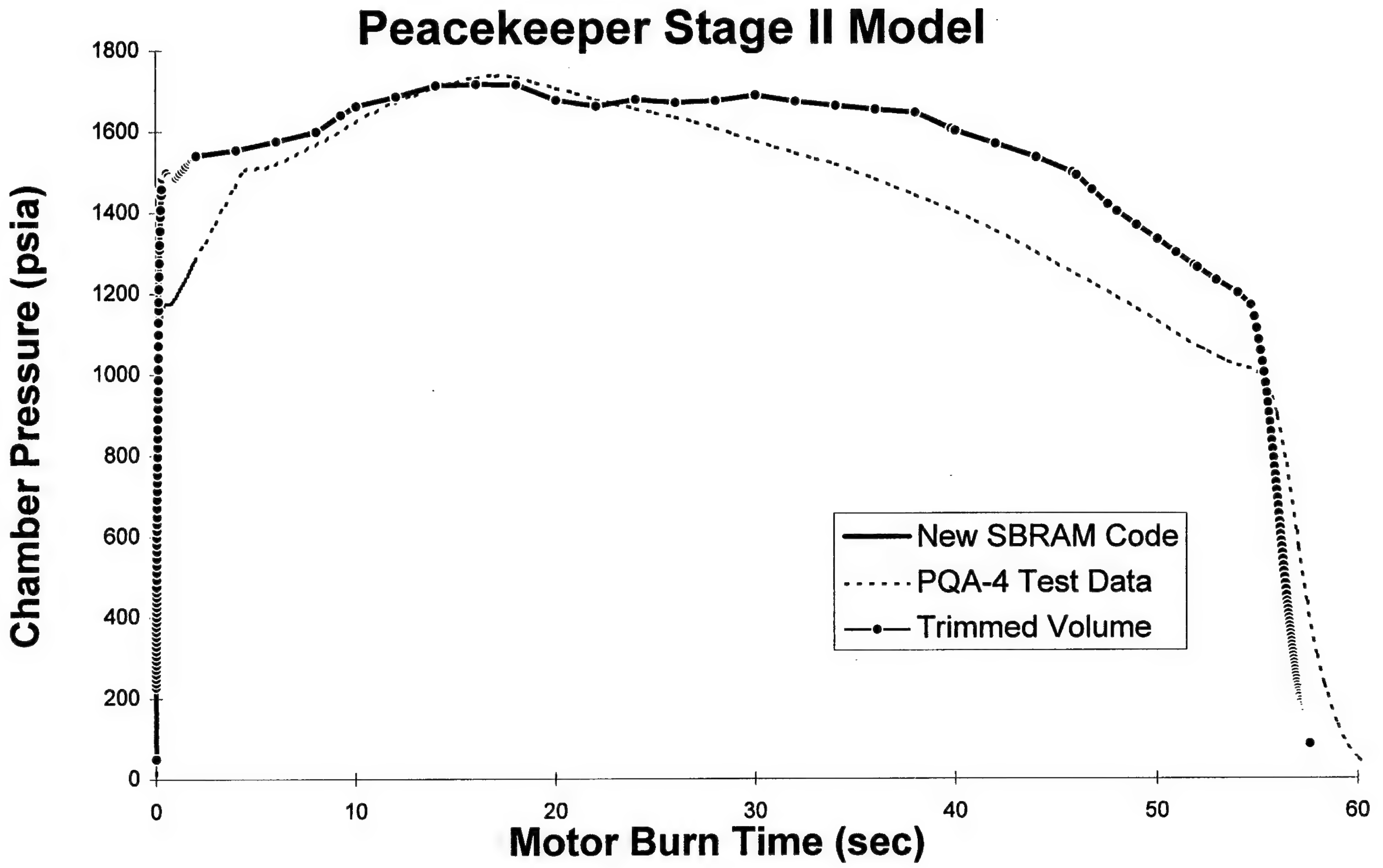


Figure 11. Comparison of the new code, test performance data, and trimmed propellant volume.

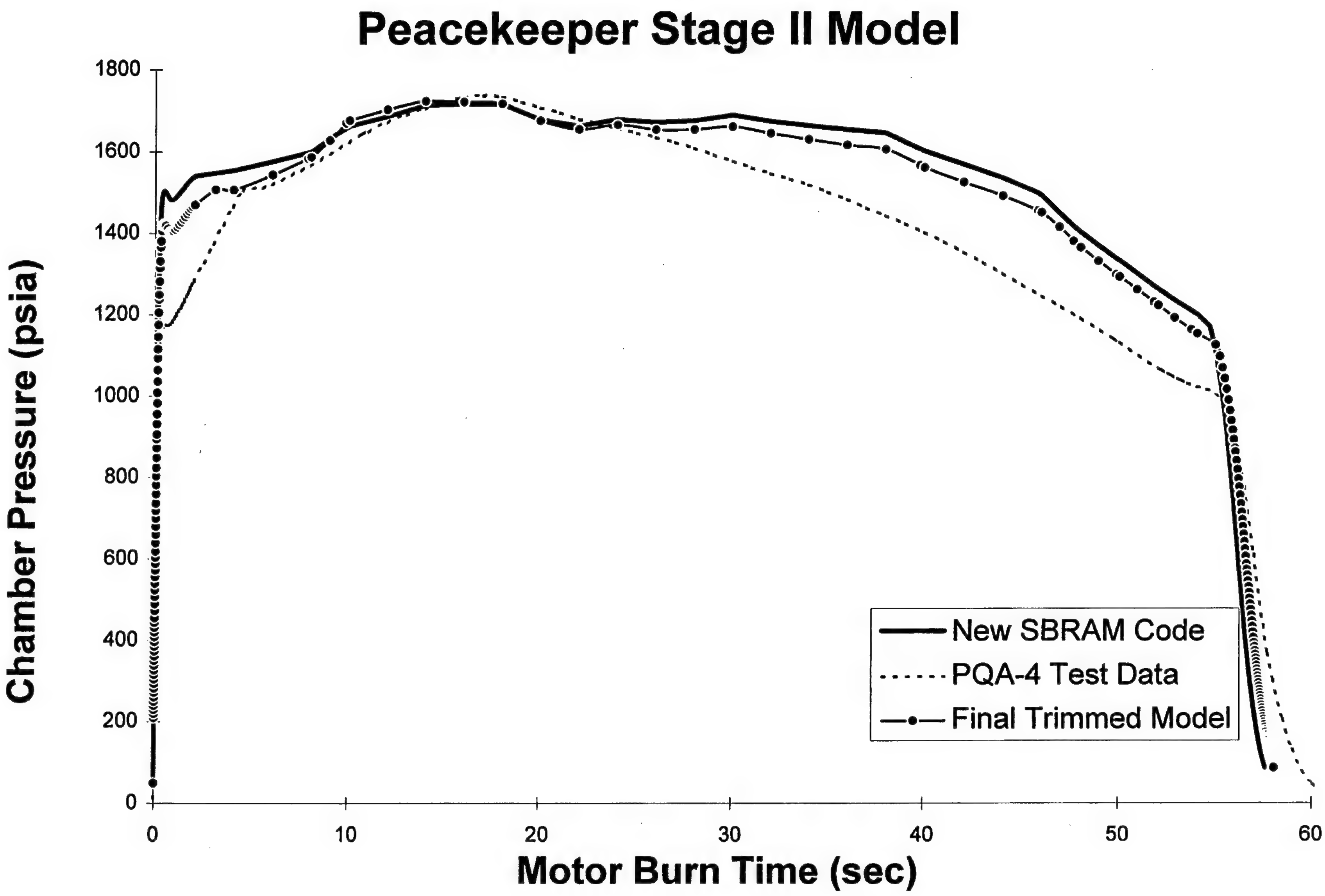


Figure 12. Comparison of the final SBRAM code, test performance data, and final trimmed model.

POWER SYSTEMS STUDIES

Eric McMahan

Coffee County Central High School
McArthur St.
Manchester, TN 37357

Final report for:
High School Apprentice Program
Power Systems Analysis

Sponsored by:
Air Force Office of Scientific Research
Boiling Air Force Base, DC

and

Arnold Engineering Development Center

August 1995

POWER SYSTEMS STUDIES

Eric McMahan
Coffee County High School

Abstract

The Power Systems Analysis team has several basic responsibilities including preventing injury, minimizing system damage, and limiting the extent of outages. Occasionally, an in depth report such as the Central Facilities Report is required. Also, in an effort to better organize information collected by the PSA team over the past five years, a pc-based index was created.

Acknowledgments

Rick McCoy

Bill Simmons

Bill Mayberry

Randy Dotson

Lance McKnight

Brian Ball

POWER SYSTEMS STUDIES

Eric McMahan

The Power Systems Analysis team is a portion of the Electrical Design section of SSI Services Inc., Arnold Air Force Base Tullahoma, Tennessee. Throughout the previous five months, the PSA team has concentrated their efforts on developing a report on the 13.8kV Central Facilities system unit substations and adjacent secondary systems of AEDC. The report serves as a guide or reference tool for future work on AEDC facilities.

The PSA team, when not involved in an in depth report such as the CF, has several basic responsibilities. It is their job to make sure the equipment here on the base is in working order and is not a danger to Air Force and civilian personnel. This also includes the minimizing of damage to system components in the event of a short circuit or other abnormality. It is also necessary to limit the extent of a power outage to a minimum amount of time as possible before getting the system running again. These objectives are met through system protection and coordination.

System protection is the detection and prompt isolation of the affected portion of a system whenever a short circuit or other abnormality occurs that might damage or adversely affect any portion of the system or the loads supplied. Coordination is the selection and/or setting of protective devices so as to isolate only that portion of the system where the abnormality occurs. Many devices are used in a protection system. Fuses and circuit breakers are two examples of these devices. A fuse is a wire or metal bar that has a very low melting point that dissolves and breaks the circuit when an electric current exceeds the specified amount. Low voltage circuit breakers contain a sensing element that, when tripped, interrupts the fault current to the system. This allows the fewest possible devices to be adversely affected by the short circuit.

Occasionally it is necessary for the PSA team to produce a report on the electrical systems of the facilities on the base. The most recent of these reports is the Central Facilities Report. The CF is a report developed to be a reference tool for future work on ADC. It is a complete evaluation of the 13.8kV Central Facilities system unit substations and adjacent secondary systems. The evaluation addressed whether the protective devices function to provide personnel safety and to minimize damage to equipment. Also, the adequacy of the system was reviewed. Whether or not the transformers, switchgear, circuit breakers, power panels, etc. can meet current requirements and can safely withstand the available fault current were the major areas addressed in the report.

The evaluation began with the unit substation primary switch and continued down through the transformer, switchgear, and circuit breakers. The evaluation ended with the first panel, motor control center, or other device immediately downstream of the unit substation switchgear. A site survey by an engineer was performed to assess the condition of the unit substation and other system components. Evaluation results and recommended corrective action (such as replacing or repairing the component) were documented.

Another project completed this summer was an index created for the PSA team in order to organize information collected over the past few years. The information organized includes the vendor data, text books, reports, software, and other general information gathered by this team. The index was preferably to be computer based were it could be easily accessable and easily updated. A database was created on Microsoft FoxPro and was used to assemble the infomation

DEMONSTRATION OF CD-ROM ARCHIVAL OF TEST INFORMATION

Laura Pickney

Franklin County High School
111 Woodbluff Rd.
Winchester, TN. 37398

Final Report for:
High School Apprentice Program
Arnold Engineering Development Center

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

and

Arnold Engineering Development Center

August 1995

DEMONSTRATION OF CD-ROM ARCHIVAL OF TEST INFORMATION

Laura Pickney
Franklin County High School

Abstract

Test information has historically been transmitted to AEDC customers in large books of computer printout, notebooks of photographs, and computer tapes of data. In order to reduce the amount of storage space required for test data, AEDC developed TPAS (Test Project Archival System) in 1994. Some of the many benefits of TPAS were that it was stored on CD-ROM, it was user-friendly with "point & click" icons, and it drastically reduced the amount of space required for data storage. It also contained more information than old-style data packages such as memos, photos, drawings, and videos. The purpose of my project was to construct a TPAS Demonstration that was publicly releasable. By copying the TPAS Demonstration onto CD-ROM and sending it around the country, potential AEDC customers will see the advantages of TPAS and doing business with AEDC.

DEMONSTRATION OF CD-ROM ARCHIVAL OF TEST INFORMATION

Laura Pickney

Acknowledgements

I would like to thank Mr. Wayne Patton, my mentor, for taking the time to teach me many things about TPAS and for helping me when I had computer problems. I would also like to thank Mr. Paul Kelly for showing me around AEDC my first week and for helping me when I had a problem. Also thanks to all the other people in the office who helped me in every way possible. I know the many things I have learned throughout the summer will help me in college and in the years to come.

General Description

Test information has historically been transmitted to AEDC customers in large books of computer printout, notebooks of photographs, and computer tapes of data. In order to reduce the amount of storage space required for test data, AEDC developed TPAS (Test Project Archival System) in 1994. Some of the many benefits of TPAS were that it was stored on CD-ROM, it was user-friendly with the "point & click" icons, and it drastically reduced the amount of space required for data storage. It also contained more information than old-style data packages such as memos, photos, drawings, and videos. Test data that once filled up a room could now be stored with TPAS (Test Project Archival System) on a 5 1/4 inch compact disc. Because TPAS was such a useful and space-saving program, potential customers needed to be shown its many benefits. The purpose of my project was to construct a TPAS Demonstration that was public releasable. By copying the TPAS Demonstration onto CD-ROM and sending it around the country, potential AEDC (Arnold Engineering Development Center) customers would see the advantages of TPAS and doing business with AEDC. Also through the TPAS Demonstration, many people would learn the easiest way to store test data. I decided, with my mentor's help, what type of information to include in the TPAS Demonstration, found examples of each type, and converted each into a file that could be displayed by TPAS. I then grouped the files in directories and sub-directories to allow the information to be separated into logical categories. Finally, I produced the Hypertext Markup Language (HTM) files used by TPAS to display the information.

Methodology

During the first week of the Summer Research Program, I familiarized myself with different aspects of the computer and the facilities at AEDC. As the second week began, my mentor and I made a plan by choosing what information we wanted included in the TPAS Demonstration. After deciding upon the type of information we wanted, we went to the SSI Public Affairs Office to check on the availability of "public release" information. The next step in my project was to set up my directory structure for the TPAS Demonstration and then set up the subdirectories to contain the items we wanted to show. The items we selected to include in the demonstration were:

1) Videos
2) Drawings
3) Spreadsheets & Graphs
4) Logs
5) Text Documents
6) Photographs
7) Slide Presentation
8) Steady State Data
9) Transient Data
10) Typical Table of Contents for Turbine Engine Tests.

I then placed "dummy" files to serve as "place-holders" inside each subdirectory so I could continue working while we obtained the actual files to include in the demonstration. I then used "TPAS Tools," an AEDC-written program, to produce text files (called the master.list) containing information about each of the files in my subdirectories. The text files contained the following information about each item to be displayed by TPAS:

1) Title of the item to be displayed on the user's computer
2) Name of the file to be displayed when the user selected that title
3) Location of the file in my directory structure
4) Type of file (text, image, etc.)

After modifying my text files to customize the titles, I used TPAS Tools to create Hypertext Markup Language files. Those would be used by TPAS to allow the user to access the different files by "pointing and clicking" with the mouse.

Next came searching for and selecting the best examples of each type of information. In each case there were specific challenges to overcome.

VIDEOS:- I found some public release videos on the AEDC Home Page on the Internet. I obtained copies and put them in the TPAS Demonstration, but it took several seconds after selecting a video before it began playing. An AEDC Computer Programmer showed me how to call another program which "launched" the video and began showing it before it all loaded from the CD into the computer memory. This eliminated the delay.

DRAWINGS: - I placed two different types of drawings into my TPAS Demonstration. The scanned drawing was scanned in with the Photostyler program, and it showed the capability of taking a full-size hand-drawn engineering drawing and storing it on CD-ROM. The electronic drawing showed the capability of storing drawings produced by Standard Software Packages such as "Autosketch".

SPREADSHEETS & GRAPHS: - I placed in the TPAS Demonstration a Chem Lab Fuel Analysis, a Chem Lab Oil Analysis, and a Fuel Analysis Summary. These spreadsheets were copied from a real TPAS. By working with the spreadsheets, I learned about Microsoft Excel.

LOGS: - I copied an electronically generated test engineer's log from a real TPAS into my TPAS Demonstration. I then changed the numbers in the log so that it would be public releasable. I also put a scanned log into the demonstration to show the varied capabilities of TPAS.

TEXT DOCUMENTS: - To show a text document, I copied a memo from a real TPAS. By working with text documents, I learned about Microsoft Word. I also scanned in a word document to show the varied capabilities of TPAS.

PHOTOGRAPHS: - There are two types of photos, normal and digital. Digital photos are taken by a camera without film and then stored on computer disk. I placed a digital photo in my TPAS Demonstration to show the capability TPAS has to display different types of photos. I also placed scanned photos as TIFF (Tag Image File Format) files and GIF (Graphic Interchange Format) files into the TPAS Demonstration. The GIF file has a built-in compression scheme that reduces the size of files while they are being transferred, while the TIFF file is an industry standard image file format supported by many applications. By putting examples of both formats, I showed the varied abilities of TPAS.

SLIDE PRESENTATION: - I placed in my TPAS Demonstration a Microsoft PowerPoint slideshow about TPAS. By working with the slide presentations, I learned about Microsoft PowerPoint. I also scanned in a slideshow about TPAS.

STEADY STATE DATA: - To be able to have steady state data in my demonstration, I had to find public releasable information. I was able to obtain steady state data from a generic turbine engine math model. (A math model is a computer program that simulates a jet engine.) The math model manufactured twenty-one data points.

TRANSIENT DATA: - I also obtained transient data from a generic turbine engine math model. The math model made two transient data points.

TYPICAL TABLE OF CONTENTS FOR TURBINE ENGINE TESTS: - I wanted to show how a typical table of contents in TPAS would look. I copied the table of contents from a typical turbine engine test to my TPAS Demonstration. By doing this I showed how compact and accessible test information can be in TPAS.

The hardware I used to complete the TPAS Demonstration was a PC, a color printer, and a scanner. During the making of the TPAS Demonstration, I used many applications on the computer. The most frequently used was TPAS Tools, which is an AEDC program for producing HTML codes. To view the TPAS Demonstration, I used Mosaic and Netscape which are Internet viewers. I also used Microsoft Word, a word processor program, Microsoft Excel, a spreadsheet program, and Microsoft PowerPoint, a presentation program to complete my project. To scan in different documents, I used the program Photostyler.

After each addition or correction that I made in my TPAS demonstration, I ran mosaic to see how it worked. Many times I had to go back and alter the master.list and then rebuild the HTM files. Through this sort of trial and error process, TPAS demonstration eventually worked.

Results

I created a TPAS demo that could be distributed on CD-ROM to any potential user to show the advantages of using TPAS. It could also be used to persuade potential customers to do business with AEDC.

References

Angell, David and Heslop, Brent. Mosaic For Dummies. IDG Books Worldwide, Inc., Chicago, 1995.

Lodholz, Steve and Bradshaw, Keith. “TPAS Tools.” 1995.

EXCERPTS FROM THE TPAS DEMONSTRATION

Attachment 1: TPAS Demonstration Home Page

Attachment 2: TPAS Demonstration Master.list

Attachment 3: Logs Page in TPAS Demonstration

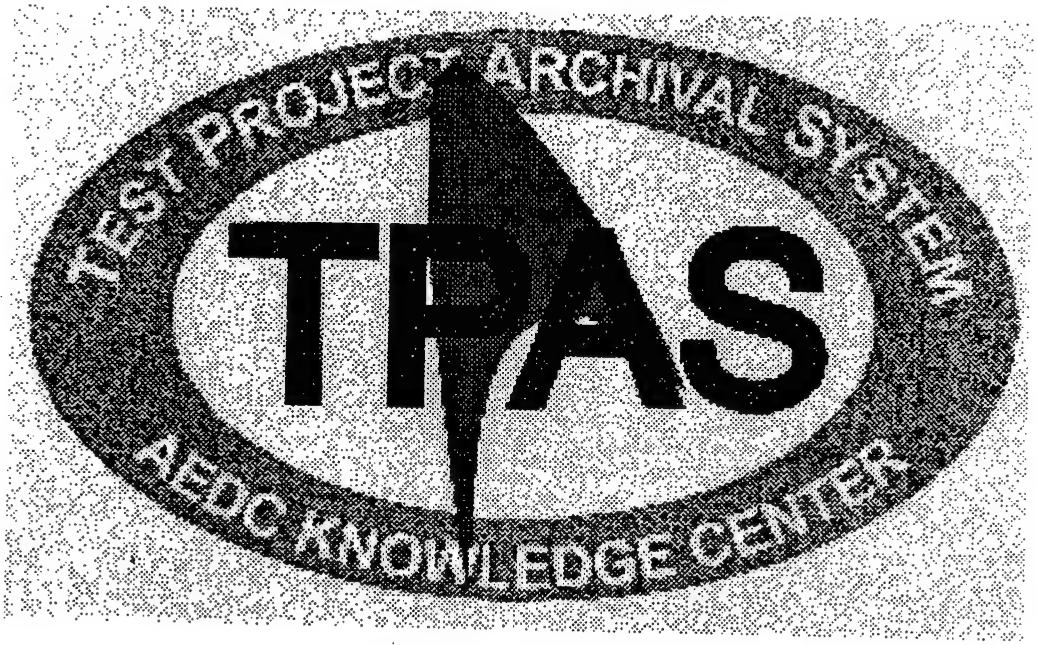
Attachment 4: Test Log in TPAS Demonstration

Attachment 5: Photographs Page in TPAS Demonstration

Attachment 6: Before Photograph in TPAS Demonstration

Attachment 7: After Photograph in TPAS

TPAS Demo



- What Is TPAS?
- Videos
- Drawings
- Spreadsheets & Graphs
- Logs
- Text Documents
- Photographs
- Slide Presentation
- Steady State Data
- Transient Data
- Typical Table of Contents for Turbine Engine Tests

```
"TPAS Demo"
21  ON
tpasclr.gif "TPAS Logo" IMAGE
tpas300.gif "logo" OFF
xxx_01.htm "What Is TPAS?" LINK
video\ "Videos" LINK
drawing\ "Drawings" LINK
tpas_pch.ppt "Slide Presentations" OFF
sprdsht\ "Spreadsheets & Graphs" LINK
logs\ "Logs" LINK
memos\ "Text Documents" LINK
photos\ "Photographs" LINK
sldshw\ "Slide Presentation" LINK
ssdata\ "Steady State Data" LINK
trdata\ "Transient Data" LINK
aedc2.htm "aedc2" OFF
link.au "link" OFF
aedc.htm "aedc" OFF
alpha\ "alpha" OFF
mock621.htm "mock621" OFF
na.htm "na" OFF
mock1\ "Typical Table of Contents for Turbine Engine Tests" LINK
xxx_01.doc "What Is TPAS?" OFF
```

Logs

- Test Engineer Log Produced Using Microsoft Excel
- Scanned Test Engineer Log
- Additional Information About Logs

TEST LOG

DATE	3/17/95	ENGINE	F99
TEST #	1	SERIAL #	FX999
JOB #	999	T. E.	Pickney/Patton

TIME	TY	No.	ALT/Mn.	PLA	T1	REMARKS
5:00						TUSL's & TAAL's SIGNED OFF
5:32						START PRETEST FLEX
5:47	SS	1				PRETEST AMBIENT
6:00						**AIR ON**
6:01						START TAPES
6:10	TR	2	9K/.6	15		ENGINE START ATTEMPT (GOOD)
6:20			30K/.9			SET 30K/.9 FLIGHT CONDITION
6:30	SS	3	30K/.9	15	STD	SS AT IDLE
6:35	TR	4	30K/.9	15-100		SNAP TO INTERMEDIATE
6:40	TR	5	30K/.9	100-15		SNAP TO IDLE
6:41	TR	6	30K/.9	15-30		ACCEL TO 30 DEG PLA
6:46	SS	7	30K/.9	30		SS AT 30 DEG PLA
6:47	TR	8	30K/.9	30-50		ACCEL TO 50 DEG PLA
6:53	SS	9	30K/.9	50		SS AT 50 DEG PLA
6:54	TR	10	30K/.9	50-70		ACCEL TO 70 DEG PLA
6:59	SS	11	30K/.9	70		SS AT 70 DEG PLA
7:00	TR	12	30K/.9	70-100		ACCEL TO INTERMEDIATE
7:05	SS	13	30K/.9	100		SS AT INTERMEDIATE
7:06	TR	14	30K/.9	100-15		DECEL TO IDLE
7:07	TR		10K/.5	15	STD	SET 10K/.5 FLIGHT CONDITION
7:25	TR	15	10K/.5	15-100-15	15	STRESS SURVEY (IDLE TO INT TO IDLE)
7:30	TR	16	10K/.5	0		ENGINE SHUTDOWN
7:40				0		**AIR-OFF**

Photographs

Scanned Photographs

- Before Data Packaging
- After Data Packaging

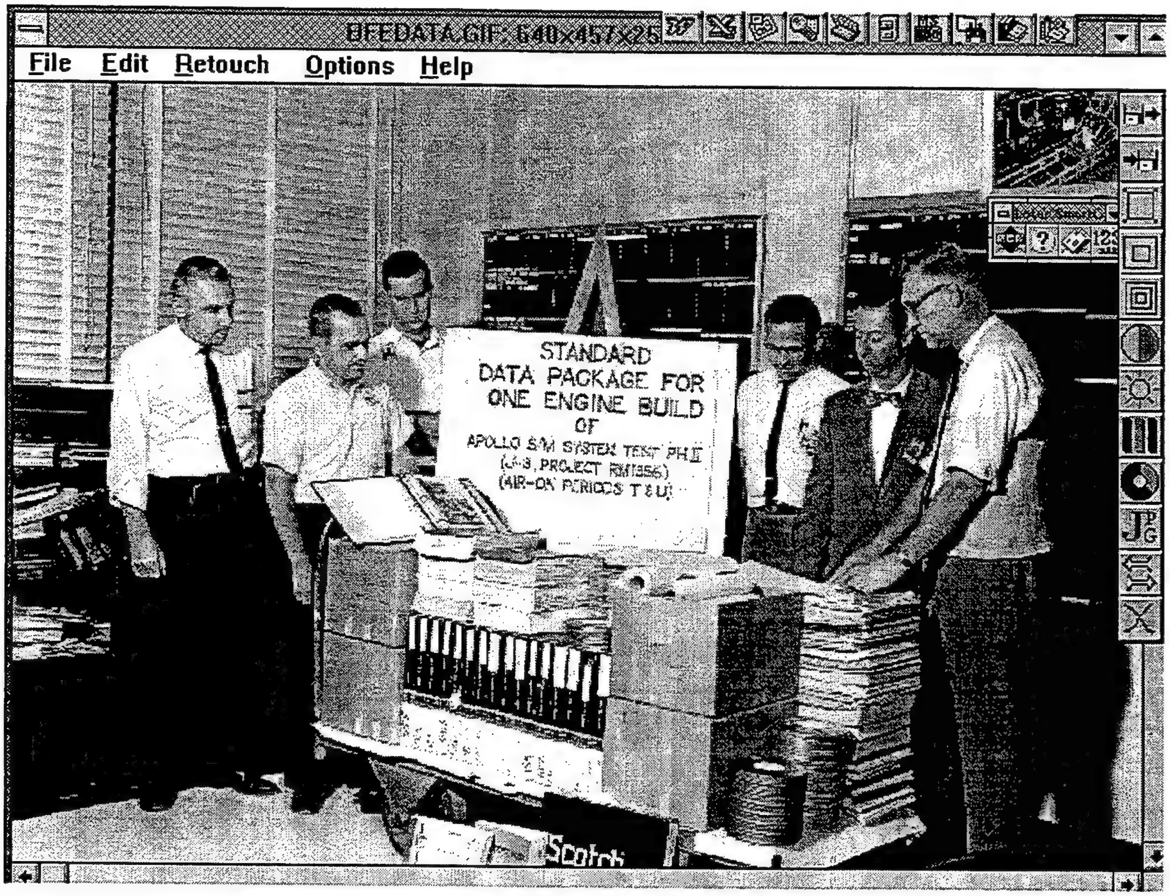
GIF Files

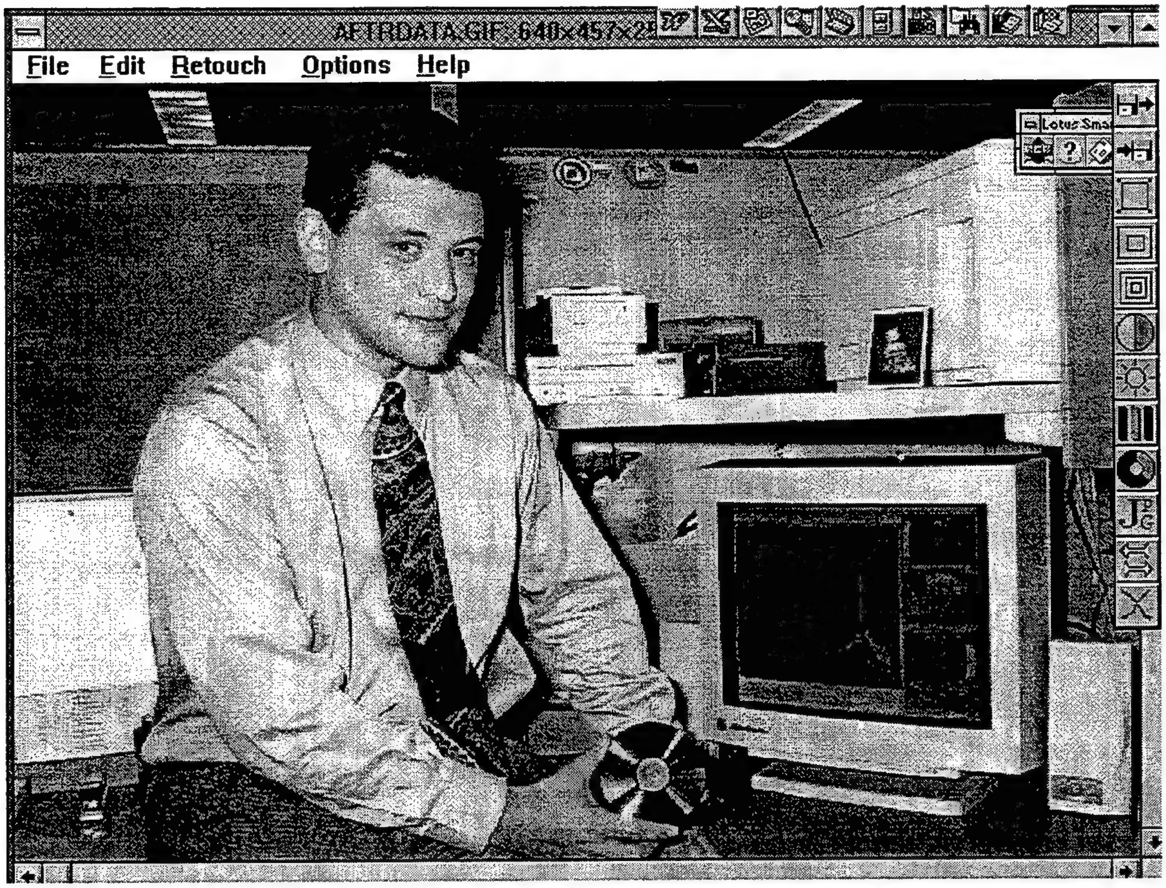
- ASTF Photograph
- Geese Photograph

TIF Files

- ASTF Photograph
- Geese Photograph

- Additional Information About Photographs


BFEDATA.GIF: 640x457x25
File
Edit
Retouch
Options
Help
STANDARD
DATA PACKAGE FOR
ONE ENGINE BUILD
OF
APOLLO S/M SYSTEM TEST PH II
(J-3, PROJECT RM1356)
(AIR-ON PERIODS T & U)
Scotch


AFTRDATA.GIF 640x457x2
File
Edit
Retouch
Options
Help
Lotus Sma

Datapoint Summary Program

Jesse Selman

Franklin County High School
925 Dinah Shore Blvd.
Winchester, TN 37398

Final Report for:
High School Apprenticeship Program
Arnold Air Force Base

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

and

Arnold Air Force Base

August 1995

DATAPOINT SUMMARY PROGRAM

Jesse Selman

Franklin County High School

Abstract

A program was written to provide the engineers at Arnold AFB's engine test facility a more rapid and efficient means of getting information about datapoints that were taken from engine tests. Each datapoint has different values for the information that is gathered during a test. The engineers need to know several different values regarding the transfer of each datapoint. The program calculates these values and prints them onto a file with the datapoints.

Acknowledgments:

I would like to thank my mentor Rusty Zarecor for his patience in teaching me a new language. I would like to thank Ron Turner for his help when Rusty was not around and for teaching me bz. Thanks to Randy Sloan for the coffee that was there to wake me up on slow mornings. I would also like to thank Mr. James D. Mitchell for his contributions to the program and his bartender jokes. I would like to thank the Air Force Office of Scientific Research for giving me this opportunity.

General Description:

During an engine test, hundreds of different types of data are gathered. These different types of data are gathered at the same time, thousands of times in each test. When this information is gathered, it is stored in a datapoint. Each datapoint is assigned a specific number. My mentor and some of his co-workers must know certain properties of each of the datapoints. I wrote a computer program that reads datapoints from a file and puts them in another file in the form of a chart. The chart lists the datapoints in numerical order from least to greatest. Listed on the chart with each datapoint is information regarding its time of duration, transferal times, CPU time, the number of records transferred, and the rate at which the records were transferred. After I debugged and tested the program, it ran very well. It will be put into use sometime next year.

DATAPOINT SUMMARY PROGRAM

Jesse Selman

In an engine test, hundreds of different values are measured at the same instant. These values are stored in a datapoint. In any one engine test, thousands of datapoints are gathered. A program called spooler transfers the datapoints from the data processing system to the data analysis system. The datapoints are listed in chronological order on a datapoint log file that contains the time that something occurred involving the datapoint, the day and date that it happened, and a three-digit code signifying what occurred. My mentor and his co-workers need to know information regarding the transferal times, CPU time, and some performance features. This information is contained in the datapoint log file. Some of the information can be read directly from the file, but some of the information must be calculated. The engineers of the Engine Test Facility needed a program, written in C to run on the desktop computers, that would find all this information and print it on a chart that could be easily read.

In order to write this program, I first had to learn the basics of C. While learning the language, I wrote several small programs that performed various functions. After one week of studying C, I began writing the datapoint summary program. I named it dpsum.c. For the next three weeks, I continued writing the program. Rusty Zarecor, my mentor, helped me whenever I had a problem that I could not solve. I had to write two subroutines for the program. One of the subroutines is named sub_times. It is for subtracting times. The other, spike_file, was to create the heading of the final output file, which I named head.dat.

The dpsum.c program first asks the user for the test cell in which the engine was tested. It then asks for the project number and the test number. It combines these three numbers into the filename of a datapoint log file(ex. 1) and opens the file. The program scans the file for different datapoints and various codes. By finding certain datapoints that have specific codes, it can find several different values. It reads the start and stop times of the transferal process directly from the file. It scans the file and uses the sub_times subroutine to find elapsed times: duration time of the datapoint, elapsed process time, elapsed performance time, and elapsed spooler time. It also reads the CPU time, number of records transferred, and rate of record transferal directly from the datapoint log file. After finding all these values, dpsum.c prints the datapoints in order from least to greatest onto a chart in the spike_file subroutine. It then prints the values for each datapoint under the correct column. (See ex. 2)

After writing dpsum.c, I debugged and tested it. The program runs successfully, and the engineers of the Engine Test Facility will start using it sometime next year.

Results:

I created an effective program that:

- Reads information from a datapoint log file
- Calculates values that are needed from a datapoint log file.
- Prints the values onto a chart with the datapoints.
- Provides the engineers of the Engine Test Facility with a quick and easy means of getting information about datapoints.

Benefits of the High School Apprenticeship Program:

Through the High School Apprenticeship Program I gained:

- A better understanding of how a computer program works.
- The ability to write a computer program in one of today's most commonly used programming languages.
- Experience working in an office environment.
- More computer usage skills.
- An idea of what becoming an engineer would require and what life as an engineer would be like.

Example 1 - Datapoint Log File

```
10:52:55  Wed 06/28/95 607
10:52:58  Wed 06/28/95 615
10:54:26  Wed 06/28/95 400
10:54:26  Wed 06/28/95 607
12:13:06  Wed 06/28/95 103 noreen
12:13:13  Wed 06/28/95 100 400001
12:13:52  Wed 06/28/95 101 400001
12:14:12  Wed 06/28/95 100 500002
12:14:17  Wed 06/28/95 402 400001 00:01:01    0.69 1
12:14:49  Wed 06/28/95 101 500002
12:14:59  Wed 06/28/95 100 400003
12:15:26  Wed 06/28/95 101 400003
12:15:35  Wed 06/28/95 100 500004
12:15:56  Wed 06/28/95 402 500002 00:01:41    1.13 1200
12:16:02  Wed 06/28/95 402 400003 00:01:00    0.68 1
12:16:17  Wed 06/28/95 101 500004
12:16:30  Wed 06/28/95 700
12:16:40  Wed 06/28/95 701
12:16:43  Wed 06/28/95 700
12:16:47  Wed 06/28/95 702
12:16:50  Wed 06/28/95 703
12:16:52  Wed 06/28/95 702
12:17:40  Wed 06/28/95 402 500004 00:01:42    1.13 1200
12:17:52  Wed 06/28/95 704 400001
12:19:39  Wed 06/28/95 705 400001 100000 10000
12:20:11  Wed 06/28/95 704 500002
12:20:34  Wed 06/28/95 704 400003
12:20:55  Wed 06/28/95 704 500004
12:21:30  Wed 06/28/95 705 400003 100000 20000
12:21:57  Wed 06/28/95 705 500002 9999999 50000
12:22:19  Wed 06/28/95 705 500004 8888889 60000
12:22:46  Wed 06/28/95 615
12:55:29  Wed 06/28/95 400
12:55:29  Wed 06/28/95 607
12:55:35  Wed 06/28/95 103 noreen
12:13:13  Wed 06/28/95 100 400005
12:13:52  Wed 06/28/95 101 400005
12:14:12  Wed 06/28/95 100 500006
12:14:17  Wed 06/28/95 402 400005 00:01:01    0.69 1
12:14:49  Wed 06/28/95 101 500006
```

Note: This example is only a small part of an actual datapoint log file.

Example 2 - Head.dat

```
                              SVERDRUP TECHNOLOGY

TITLE  : EDPS DATA POINT HISTORY PRINT                 LOG FILENAME : process-control-
c22356-rz01-log
Project:  2356
TEST   :  rz01
DATE   :  Fri Jul 14 10:23:04 1995


DP        DP      START       END       ELAPSED     << PERFORMANCE >>
SPOOLER
NO.       DUR    PROCESS     PROCESS    PROCESS     ELAPSED   CPU   N.REC   RATE

400001   00:39   12:13:52   12:19:39   00:05:47    00:01:01  0.690      1   0.00     01:47
500002   00:37   12:14:49   12:21:57   00:07:08    00:01:41  1.130   1200  32.00     01:46
400003   00:27   12:15:26   12:21:30   00:06:04    00:01:00  0.680      1   0.00     00:56
500004   00:42   12:16:17   12:22:19   00:06:02    00:01:42  1.130   1200  28.00     01:24
400005   00:39   12:13:52   12:19:39   00:05:47    00:01:01  0.690      1   0.00     01:47

500006   00:37   12:14:49   12:21:57   00:07:08    00:01:41  1.130   1200  32.00     01:46
400007   00:27   12:15:26   12:21:30   00:06:04    00:01:00  0.680      1   0.00     00:56
500008   00:42   12:16:17   12:22:19   00:06:02    00:01:42  1.130   1200  28.00     01:24
400009   00:39   12:13:52   12:19:39   00:05:47    00:01:01  0.690      1   0.00     01:47
500010   00:37   12:14:49   12:21:57   00:07:08    00:01:41  1.130   1200  32.00     01:46

400011   00:27   12:15:26   12:21:30   00:06:04    00:01:00  0.680      1   0.00     00:56
500012   00:42   12:16:17   12:22:19   00:06:02    00:01:42  1.130   1200  28.00     01:24
400013   00:39   12:13:52   12:19:39   00:05:47    00:01:01  0.690      1   0.00     01:47
500014   00:37   12:14:49   12:21:57   00:07:08    00:01:41  1.130   1200  32.00     01:46
400015   00:27   12:15:26   12:21:30   00:06:04    00:01:00  0.680      1   0.00     00:56

500016   00:42   12:16:17   12:22:19   00:06:02    00:01:42  1.130   1200  28.00     01:47
500018   00:42   12:16:17   12:21:57   00:05:40    00:01:42  1.130   1200  32.00     01:46
400019   00:27   12:15:26   12:21:30   00:06:04    00:01:00  0.680      1   0.00     00:56
500020   00:37   12:14:49   12:22:19   00:07:30    00:01:41  1.130   1200  28.00     01:24
400021   00:39   12:13:52   12:19:39   00:05:47    00:01:01  0.690      1   0.00     01:47

500022   00:37   12:14:49   12:22:19   00:07:30    00:01:41  1.130   1200  32.00     01:24
400023   00:27   12:15:26   12:21:30   00:06:04    00:01:00  0.680      1   0.00     00:56
500024   00:42   12:16:17   12:21:57   00:05:40    00:01:42  1.130   1200  28.00     01:46
400025   00:39   12:13:52   12:19:39   00:05:47    00:01:01  0.690      1   0.00     01:47
500026   00:37   12:14:49   12:22:19   00:07:30    00:01:41  1.130   1200  32.00     01:24
```

Note: This example is only part of the actual file.

FLOW CALIBRATION SYSTEMS MODIFICATION

Ross Stevenson

Coffee County Central High
2001 McArthur
Manchester, TN 37355

Final Report for:
High School Apprentice Program

Sponsored by:
Air Force Office of Scientific Research
Bolling Air Force Base, DC

and

Arnold Engineering Development Center
Bionetics Corporation

Dexter Shelton

August 4, 1995

FLOW CALIBRATION SYSTEMS MODIFICATION

Ross Stevenson
Coffee County High School

Abstract

Flow meter calibration on multiple fuels requires the ability to fill and purge flow lines and calibration systems. The integrity of the calibration is dependent on the ability to identify the consistency of the fluid used in the calibration. Proper placement and adequate arrangement of valves expedites the process of changing fuels.

FLOW CALIBRATION SYSTEMS MODIFICATION

Ross Stevenson

Acknowledgments

I would like to thank my mentor, Dexter Shelton, and Charlie Fagg and James Mitchell for all of their support and guidance during my summer tour.

Introduction

The Bionetics Corporation is purchasing a new liquid flow calibration bench. When this new bench arrives, the Environmental Flow Calibrator and the Cox Flow Calibrator will be removed. After their removal, the FMSI (Flow Measurement System Incorporated) bench will be relocated to the former location of the Environmental Flow Calibrator. The proper water and fuels supply piping and draining facilities will be installed, and a new valve system will be designed to accommodate the new configurations.

Methodology

My project is to design the plumbing and layout of the new liquid flow calibration equipment. The new flow calibration system (bench) requires water and fuels supply piping, drainage facilities, and electrical power. The layout will be used by electricians and pipefitters to install the new plumbing and electrical lines.

Each bench requires plumbing for JP8, JP4, drain, pressure, and water. Plumbing for each bench must be independent so there will be no interference during operation. Each bench requires spill provisions to contain any fuel leaks or spills. The first thing I did was measure the dimensions of the Dynamic Flow Building and the flow benches. From the dimensions of the flow benches I calculated the dimensions of the spill trays to contain the amount of fuel in each flow system. I designed the trays with the required dimensions and a 1/2 in. tubing in the side of the tray for draining. Next, I mapped out the flow building to determine the best location for the FMSI and new flow benches. I examined the existing fuel, pressure, drain, and water piping to determine which routes to use in my layout. I then designed the plumbing system. In addition, I determined both numbers and types of valves required to meet the tasks outlined to me by my supervisor. From various valve catalogs, I chose the valves needed for my design. I chose three-way 1/2 in., female NPT, valves for each bench to select JP8 or JP4 fuel. My plans call for four of these valves: one each, for the FMSI and Flow Tech, and two, for the new bench. The cost of these valves is $139.10 each. I chose a one-way ball valve to select the bench to supply fuel to. Seven of these valves are required in my design. The valves cost $147.20 each. I selected a 1/2 in., female NPT poppet check valve to drain each bench into one fuel container without any back fill into the other benches. These valves cost $91.60 each. My design calls for three of them. I have included an additional fuel pump in the fuel storage building as part of my design. This pump provides the technician with ready access to a different fuel when changing fuels. The pump that I selected costs $891. The total costs of valves and pump are $2,752.60.

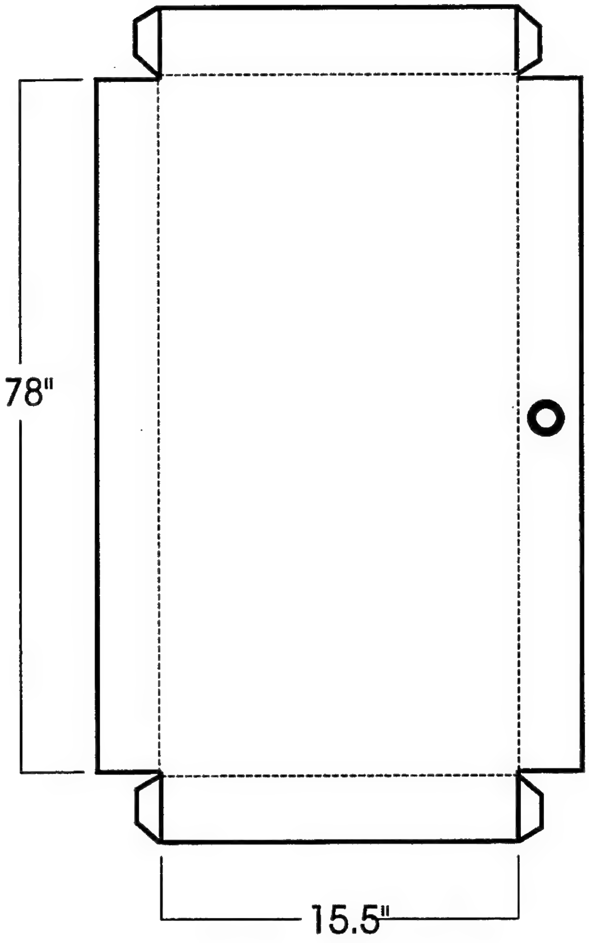
There will be approximately 200 feet of plumbing required. All tubing will be 1/2 in. o.d. stainless steel. I recommend a section of vinyl tubing at each bench for visual inspection of fuel drainage. This will visually indicate when the bench is completely drained. This is a necessity on the FMSI bench and will help on the Flow Tech and possibly, the new bench. The approximate amount of electrical conduit is 40 feet.

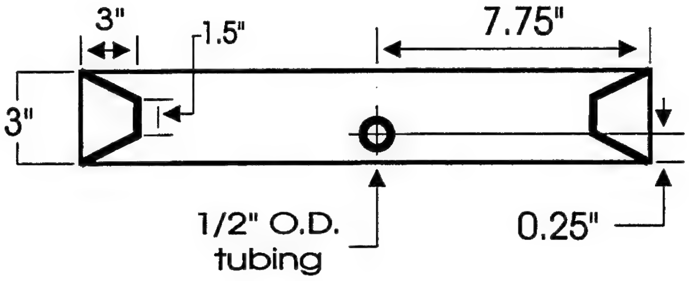
Results

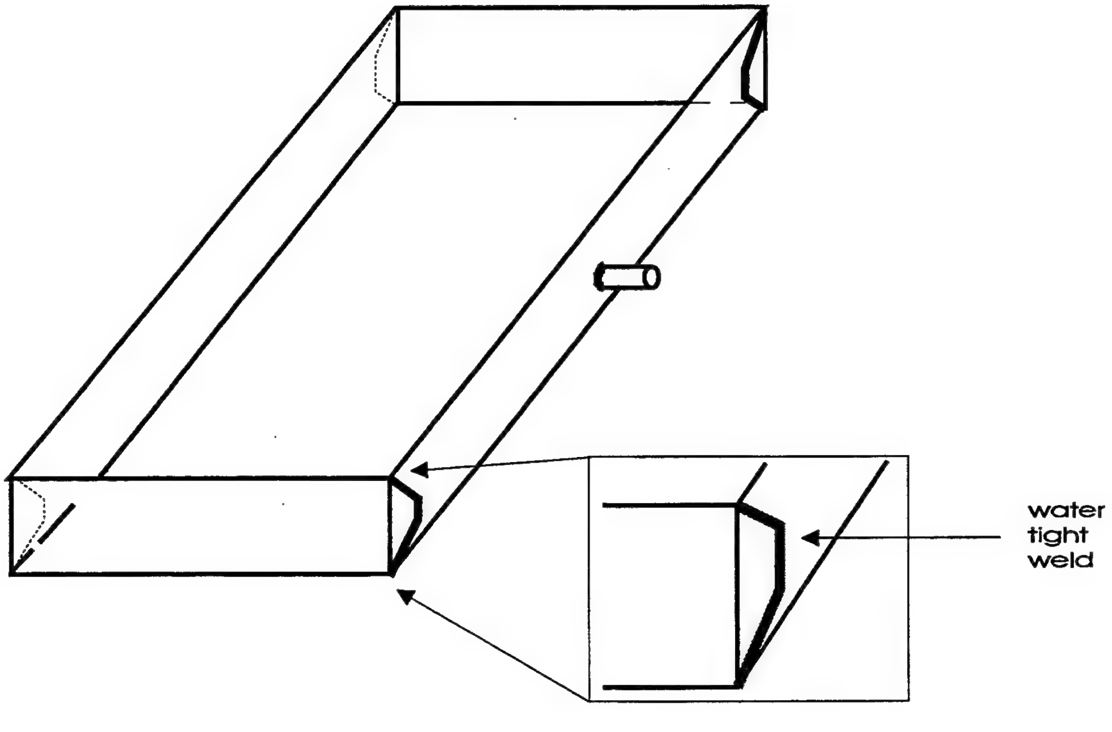
During my summer research tour, I designed spill provisions for the FMSI and Flow Tech, the valve system for the Flow Dynamics Building, and the plumbing layout for JP8 and JP4, drain system, and water. I submitted the drawings for valve modifications and plumbing layouts design. I also submmitted the drawings for fabrication of the spill pans.

I learned the characteristics of flow calibration and how to construct engineering drawings. I had the opportunity to make purchasing decisions based on quality versus price. I also had a chance to work with other engineers and see how they deal with different situations.

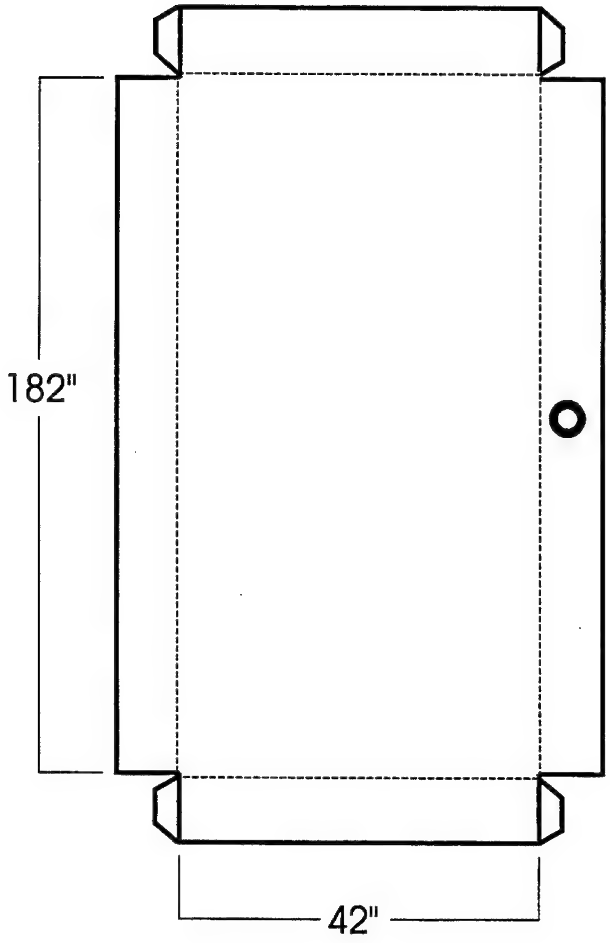
FMSI

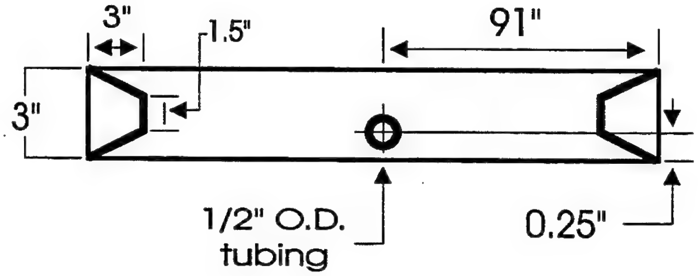


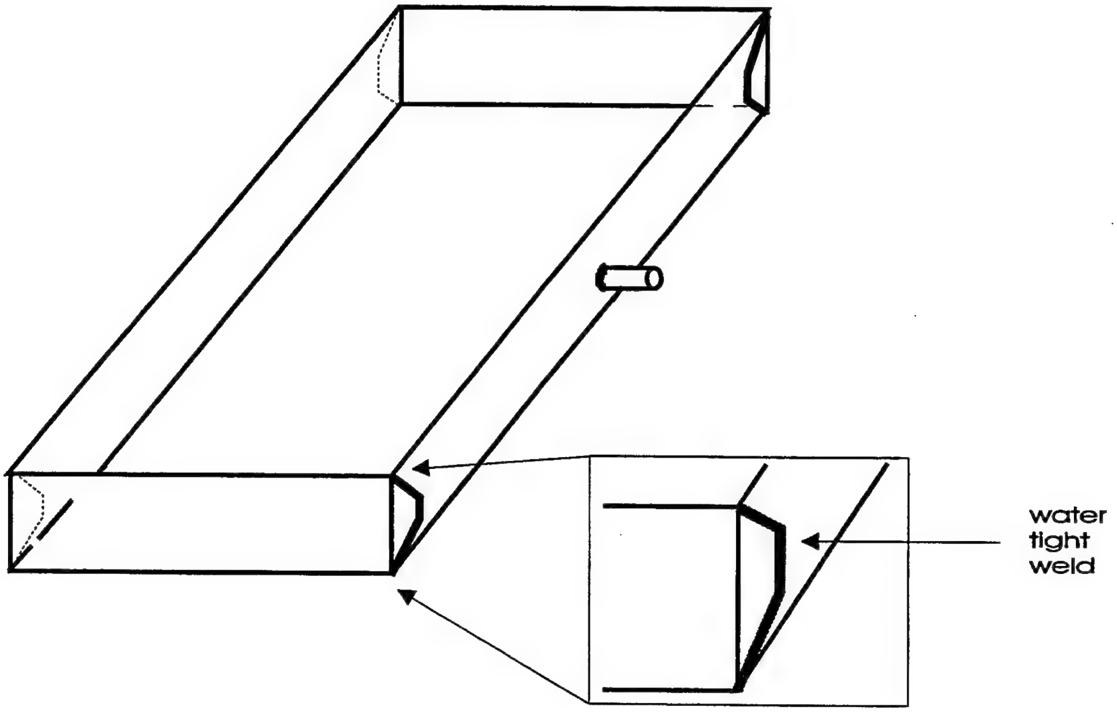




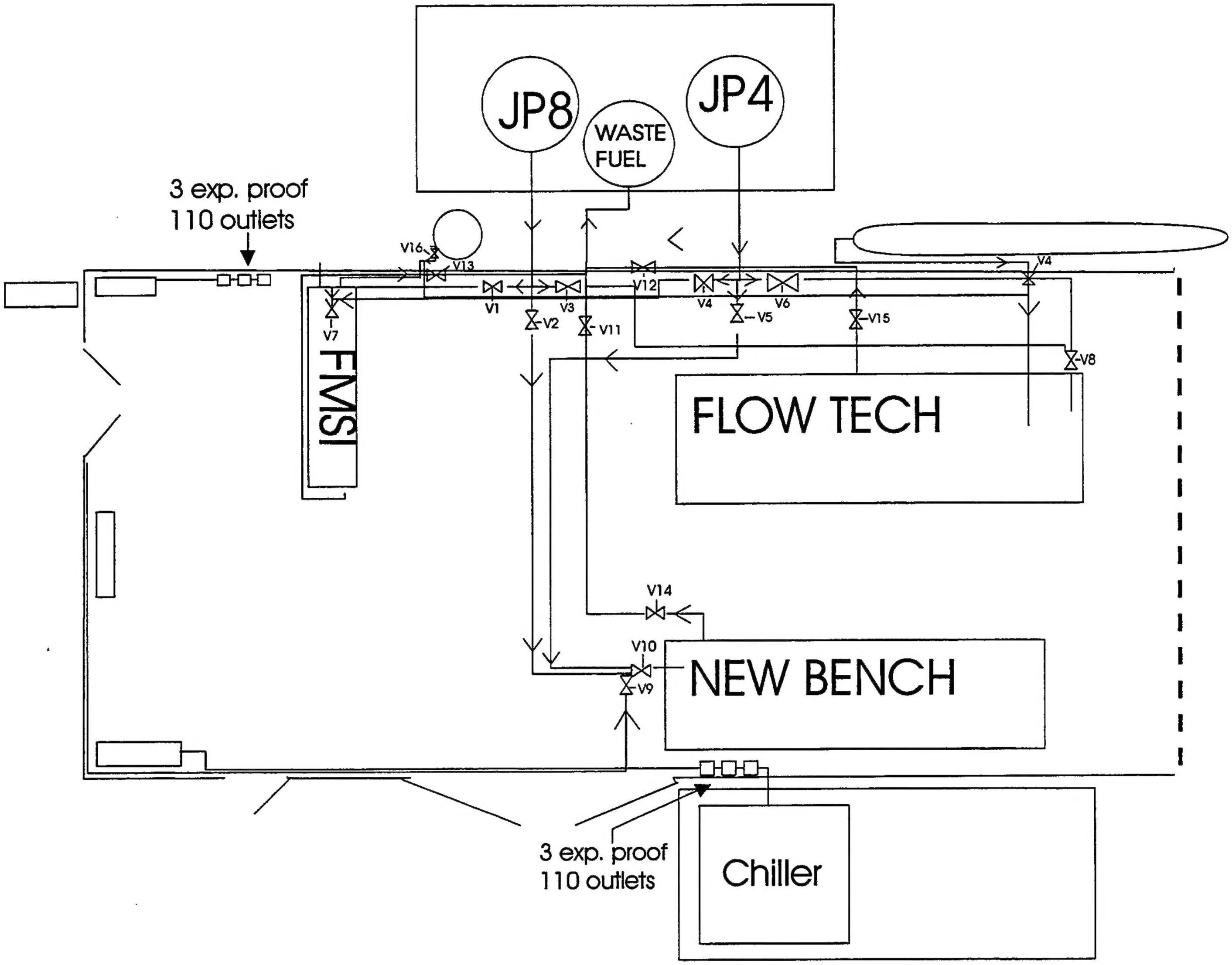
Flow Tech







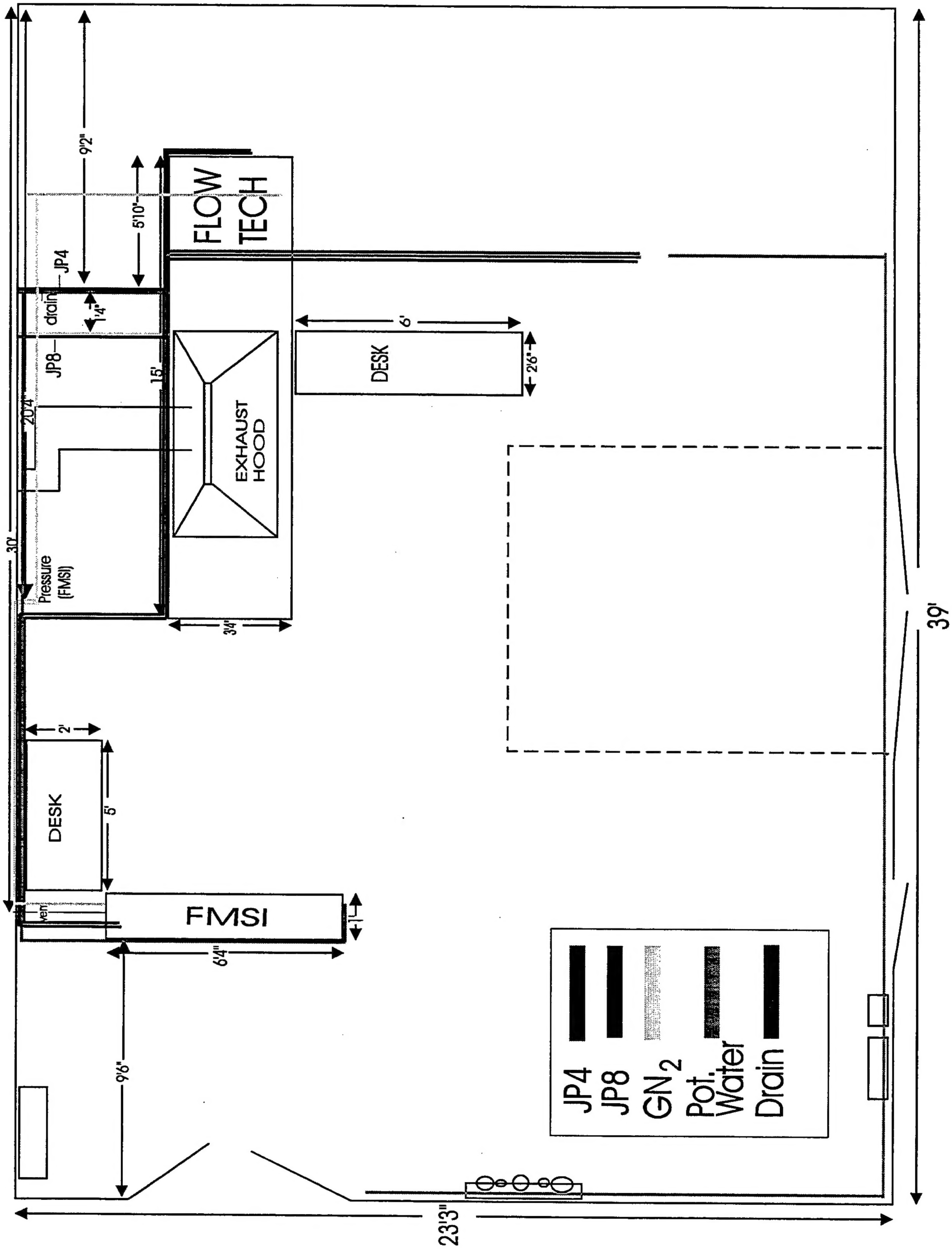
Block Diagram of Proposed Flow Configuration

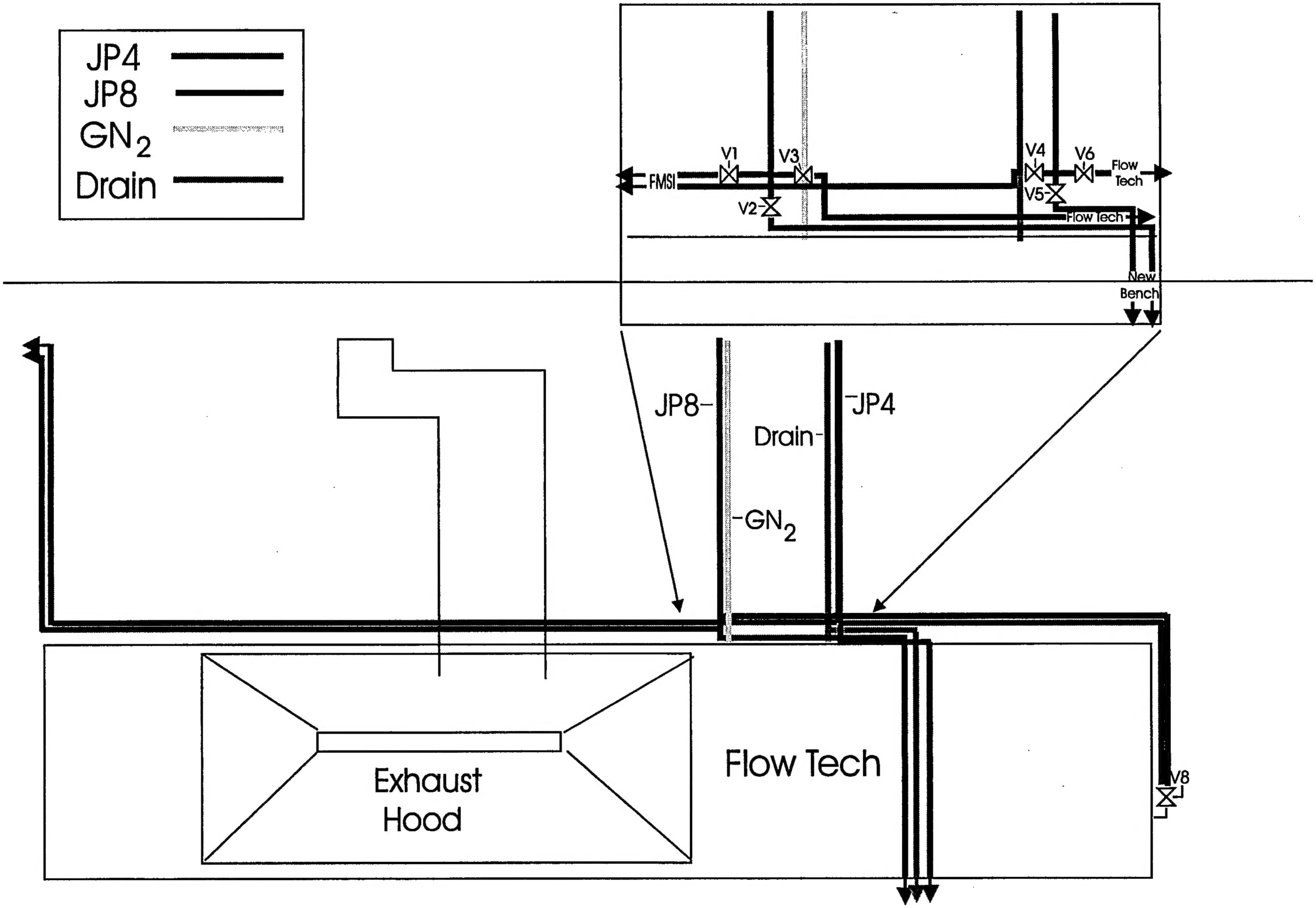


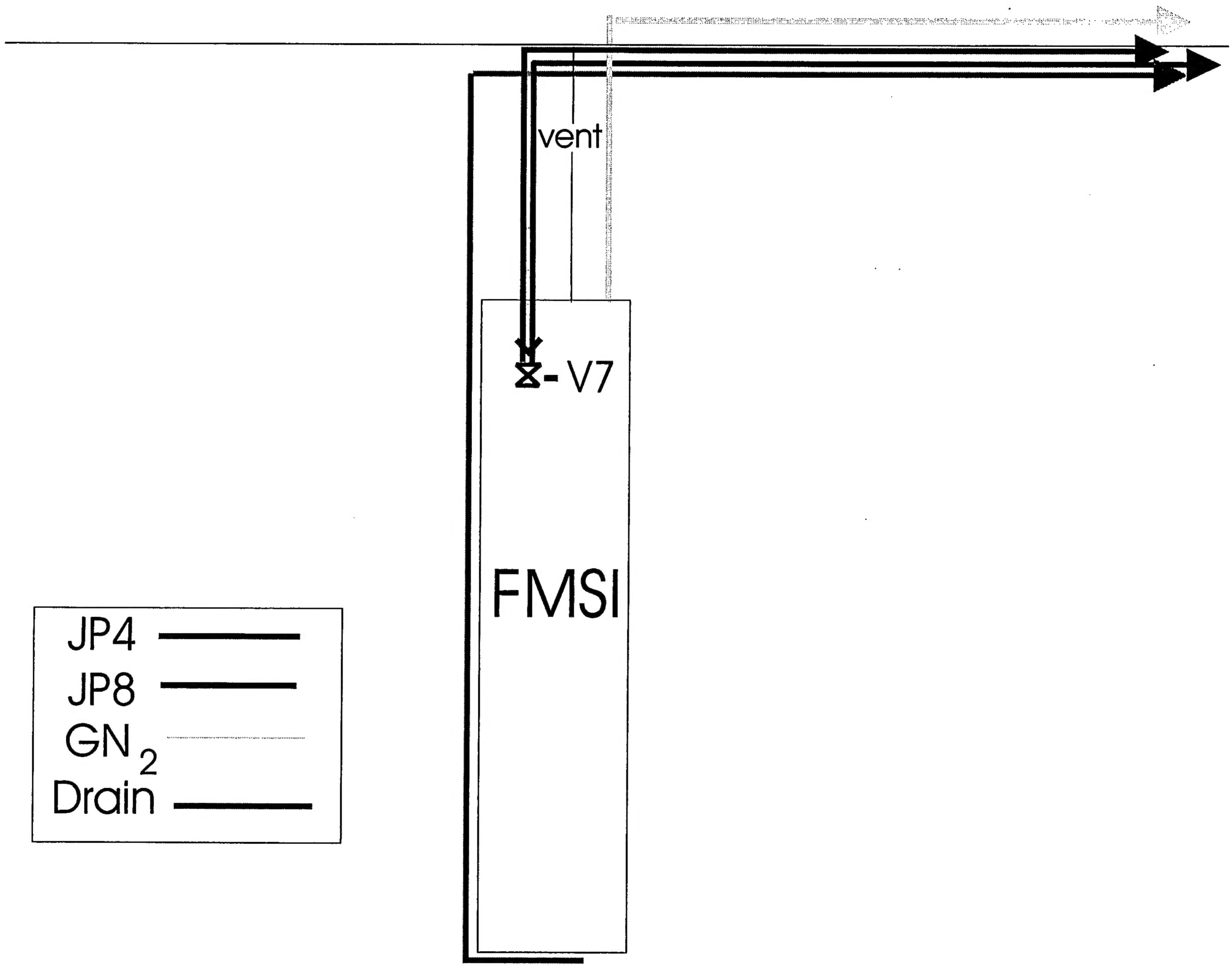
V1-V6: 1-way ball valve
V7-V10 : 3-way ball valve
V11-V14 :Poppet-Check Valve
V15-V16 : Exsisting valve

*All fuel lines 1/2 in. o.d. stainless steel tubing

Main layer - 1
Plumbing -2
Pressure-3
Electrical-4
Valves-5


FLOW TECH
9'2"
5'10"
JP4
drain
1'4"
JP8
15'
20'4"
30'
Pressure (FMSI)
EXHAUST HOOD
3'4"
6'
DESK
2'6"
39'
2'
DESK
5'
vent
FMSI
1'
6'4"
9'6"
23'3"
JP4
JP8
GN2
Pot. Water
Drain


JP4
JP8
GN$_2$
Drain
V1
V3
V4
V6
V2
V5
FMSI
Flow Tech
Flow Tech
New Bench
JP8
JP4
Drain
GN$_2$
Flow Tech
Exhaust Hood
V8


vent
V7
FMSI
JP4
JP8
GN_2
Drain